Century Data Systems
A Xerox Company

Marksman

Marksman T Series Disk/Tape Formatter

Performance Specification

M20, M40, M80, M160.

Part No. 76235-900

Marksman T Series
Disk/Tape Formatter

Performance Specification

M20, M40, M80, M160.

September 1981

Marksman specifications are subject to change without notice.

Century Data Systems
A Xerox Company

RECORD OF REVISIONS	
REVISION	**NOTES**
0	ORIGINAL ISSUE – Sept. 1981

TABLE OF CONTENTS

Section		Page
1	INTRODUCTION	1-1
	1.1 General	1-1
	1.2 Marksman T Series Features	1-1
	1.3 General Description	1-1
2	PHYSICAL DESCRIPTION	2-1
	2.1 Size	2-1
	2.2 Connectors	2-1
	2.3 Mounting	2-1
	2.4 Power Consumption	2-1
	2.5 Environmental Limits	2-1
3	HOST INTERFACE	3-1
	3.0 General	3-1
	3.1 Interface Architecture	3-1
	3.1.1 Command Register	3-1
	3.1.2 Status Register	3-1
	3.2 Interface Signal Description	3-3
	3.2.1 Signal Levels	3-3
	3.2.2 Signal Terminations	3-3
	3.2.3 Signal Loading	3-4
	3.2.4 Signal Definition	3-4
	3.3 Timing	3-5
4	COMMAND SET	4-1
	4.0 General	4-1
	4.1 Command Sequence (Single Sector Operation)	4-1
	4.2 Multiple Sector Access	4-1
	4.3 Direct and Transparent Commands	4-1
	4.4 Disk Commands	4-2
	4.4.1 Boot	4-4
	4.4.2 Sequence Drive Up	4-4
	4.4.3 Sequence Drive Down	4-5
	4.4.4 Format Command	4-5
	4.4.4.1 Format Track With No Verify	4-5
	4.4.4.2 Format Track With CRC Verify	4-5
	4.4.4.3 Format Track With FIFO Data and No Verify	4-6
	4.4.4.4 Format Track With FIFO Data and CRC Verify	4-6
	4.4.5 Read	4-7
	4.4.5.1 Read Sector With Retry	4-7
	4.4.6 Write	4-7
	4.4.6.1 Write Sector	4-7
	4.4.7 CRC Track Verify	4-8
	4.4.8 Read Drive Status	4-8
	4.4.9 Seek	4-9
	4.4.10 Drive Identification	4-9
	4.5 Tape Commands	4-10
	4.5.0.1 Tape Read/Write Modes	4-11
	4.5.0.2 Tape Availability	4-11
	4.5.1 Position	4-12
	4.5.2 Write Data	4-13
	4.5.3 Write File Mark	4-14
	4.5.4 Read Data	1-14
	4.5.5 Advance File Mark(s)	4-14
	4.5.6 Read Tape Status	4-15
	4.6 Disk/Tape Commands	4-15
	4.6.1 Burst Disk Back-Up	4-15

CONTENTS (Continued)

Section		Page
4	4.6.2 Transparent Disk Back-Up	4-16
	4.6.3 Burst Disk Restore	4-17
	4.6.4 Transparent Disk Restore	4-17
	4.7 System Commands	4-18
	4.7.1 No-Op	4-18
	4.7.2 Command Acknowledge	4-19
	4.7.3 Read Transparent Command Status	4-19
	4.7.4 Transparent Command Acknowledge	4-19
	4.8 Test Commands	4-19
	4.8.1 Memory Dump	4-19
	4.8.2 Host FIFO Fill	4-20
	4.8.3 Host FIFO Access Test	4-20
	4.8.4 Command/Status Register Test	4-20
	4.8.5 Formatter Identification	4-21
5	MARKSMAN DISK INTERFACE	5-1
	5.0 General	5-1
	5.1 Standard Marksman Interface Connector	5-1
	5.2 Sector Size	5-1
	5.3 Disk Defect Map	5-1
6	ARCHIVE TAPE INTERFACE	6-1
	6.0 General	6-1
	6.1 Archive Connector	6-1
	6.2 Data Block Size	6-1
	6.3 Tape Positioning	6-1
7	DEI TAPE INTERFACE	7-1
	7.0 General	7-1
	7.1 DEI Connector	7-1
	7.2 Data Block Size	7-1
	7.3 Tape Positioning	7-1

Appendix		Page
A	MARKSMAN DISK DRIVE SPECIFICATIONS	A-1
	A.1 Unformatted Drive Specifications	A-1
	A.2 Formatted Drive Parameters	A-1
B	T SERIES FORMATTER DISK FORMAT	B-1
	B.1 Sector Format	B-1
	B.2 Defective Media Recovery	B-1
	B.2.1 Defective Media Map	B-2
	B.2.2 Defective Map Creation Logic	B-2

ILLUSTRATIONS

Figure		Page
1-1	T Series Formatter Block Diagram	1-2
2-1	T Series Formatter Physical Configuration	2-1
3-1	Interface Signal Set	3-4
3-2	Status Register Read Timing	3-5
3-3	Command Register Write Timing	3-6
3-4	FIFO Read Timing	3-6
3-5	FIFO Write Timing	3-7
B-1	Hard Sector Embedded I.D. Format	B-1
B-2	DRVFMT Flowchart (Page 1 of 6)	B-4
B-2	DRVFMT Flowchart (Page 2 of 6)	B-5
B-2	DRVFMT Flowchart (Page 3 of 6)	B-6
B-2	DRVFMT Flowchart (Page 4 of 6)	B-7
B-2	DRVFMT Flowchart (Page 5 of 6)	B-8
B-2	DRVFMT Flowchart (Page 6 of 6)	B-9

TABLES

Table		Page
3-1	Command Register (CMDREG) Definition	3-2
3-2	Status Register (STREG) Definition	3-2
3-3	Formatter Status Byte Definition (STREG + 0)	3-3
3-4	Host Connector Pin Assignments	3-3
4-1	Disk Commands	4-2
4-2	Command Register Definition for Disk Commands	4-3
4-3	Status Register Definition for Disk Commands	4-3
4-4	Disk Command Completion Codes	4-4
4-5	Marksman Status Bytes	4-9
4-6	Marksman Drive ID Codes	4-10
4-7	Sector Size Codes	4-10
4-8	Tape Commands	4-10
4-9	Command Register Definition for Tape Commands	4-10
4-10	Status Definition for Archive Tape Commands	4-11
4-11	Status Register Definition for DEI Tape Commands	4-12
4-12	Tape Command Completion Codes	4-12
4-13	Tape Read/Write Mode Control Logic	4-13
4-14	Tape Mode/Command Protocol	4-13
4-15	Disk/Tape Commands	4-15
4-16	Disk/Tape Command Completion Codes	4-16
4-17	System Commands	4-18
4-18	System Command Completion Codes	4-18
5-1	Marksman Connector Pin Assignments	5-1
5-2	Secondary Drive Interface Connector	5-1
6-1	Archive Connector Pin Assignments	6-1
7-1	DEI Connector Pin Assignments	7-1
A-1	Marksman Media Parameters	A-1
A-2	Formatted Data Capacity	A-1
B-1	Block Definition	B-1
B-2	Defective Media Memory Map	B-3

SECTION 1
INTRODUCTION

1.1 GENERAL

The Century Data Systems T series disk/tape formatter provides, on one PC card, a complete Marksman disk controller, ARCHIVE Cipher Data or Data Electronics Inc. tape controller, and simple, but powerful host interface. By combining these features on one card, it creates three major advantages for the host system: dramatically increased data transfer rates, off-line back-up and restore capability, and simplicity in the host interface.

First, the T series can support very high data rates to the host, up to 2.0 Mbytes/s burst and 500 Kbytes/s continuous. Second, the formatter offers true off-line data back-up and restoration through its "transparent" commands. Even with a previously commanded "transparent" back-up or restore operation in progress, the host can still access a disk. Third, the simple host interface requires only eight data lines, four address lines, and three control lines (READ, WRITE, and RESET). In addition, this formatter provides all signals necessary to support DMA and other control features for high performance systems.

1.2 MARKSMAN T SERIES FEATURES

1. Controls up to two Marksman disk drives (any mix of M20, M40, M80, and M160 Models).
2. Controls up to four ARCHIVE Cipher Data or Data Electronics Inc. streaming tape drives (both 10 and 20 Mbyte).
3. Simple parallel port interface to host computer.
4. Supports DMA rates of 2.0 Mbytes/s burst and 500 Kbytes/s continuous.
5. CRC error detection on disk data transfers.
6. Multiple sector access capability.
7. Automatic compensation for disk and tape media defects.
8. Supports overlapping seeks.
9. 10 Mbyte back-up in less than two minutes.
10. Off-line, "transparent" back-up allows continued host access to disk and tape (can be prioritized).

1.3 GENERAL DESCRIPTION

The T series formatter has six basic functional blocks, as shown in Figure 1-1, the T Series Formatter Block Diagram. The six blocks include: 1) on-board micro-controller, 2) serial data sequencer, 3) three floating data buffers, 4) host interface, 5) disk interface, and 6) tape interface.

The on-board micro-controller includes a Z-80 CPU, EPROM, RAM, and I/O control circuitry. It reads the command register, initiates host commands, and updates the status register at command completion. Due to the very high data rate capability of the T series the micro-controller does not actually do any data transfers, but merely initiates the transfer sequences through its I/O circuitry.

The serial data sequencer performs serial-parallel data conversion for disk data and also does the CRC error check on the serial data stream. It works together with the disk interface to handle all disk data transactions.

The micro-controller handles all data transactions with the three floating data buffers. It can assign them individually to any of the host, disk, or tape data busses, to execute a data transfer in "bucket brigade" fashion. First, it assigns an empty buffer to the source device. When the buffer is full, the micro-controller assigns it to the destination device and assigns the next empty buffer to the source device. The buffers rotate in this fashion, full ones from source to destination device, and empty ones back in a rotary fashion, until the transfer is complete.

The three interface blocks generate the necessary interface signals for the disk, tape and host. Also, they provide terminations for inputs and drivers for outputs. Each interface includes a data bus, address bus, and control signals. This manual does not describe the disk and tape interfaces in detail, but Section 3.0 gives a thorough description of the host interface.

1.4 RELATED DOCUMENTS

Companion documents that are available include:

Marksman Performance Specification	P/N 76220-9XX
Marksman Disk Drive Technical Manual	P/N 76220-1XX
Marksman Installation Manual	P/N 76220-2XX

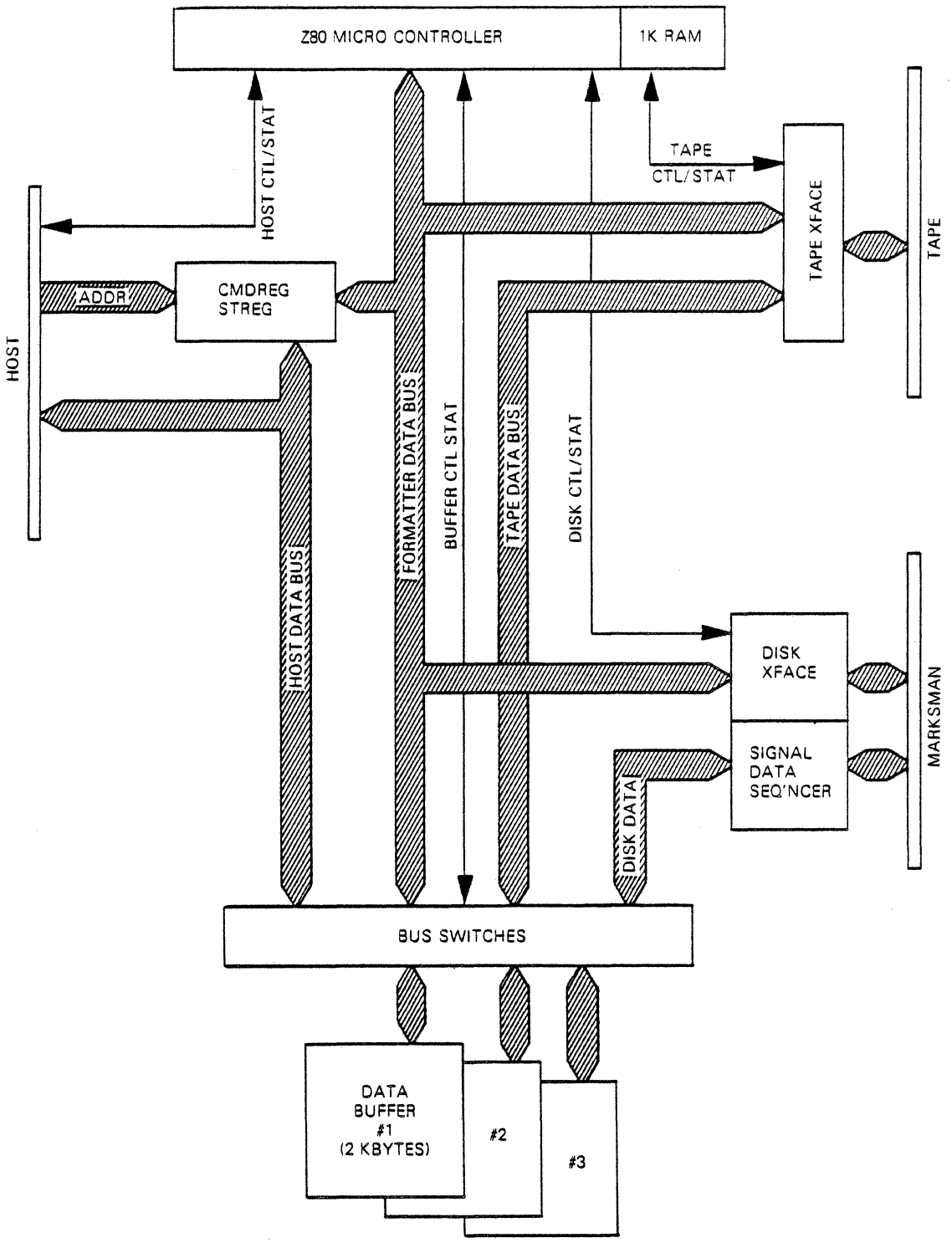


Figure 1-1. T Series Formatter Block Diagram

SECTION 2
PHYSICAL DESCRIPTION

2.1 SIZE

The Formatter card is 11.5" by 16.00" and is .062" thick. With components, the assembled card is 0.70" thick, and requires 0.10" clearance on each side. Figure 2-1 shows the physical configuration of the PC card.

2.2 CONNECTORS

The T series formatter has five connectors, designated J1 – J5. J1, the host ribbon cable connector, is a 40 pin connector, with mating plug part no. 51-1140-11 (Winchester Products IDC socket connector with strain relief). J2, the primary disk drive ribbon cable connector, is a 40 pin connector, as is J1, the host ribbon cable connector. This connector mates with the Marksman interface connector, via a 4" (max. length) cable, on the host disk drive. (The formatter physically mounts to the primary disk drive.) J4 is the secondary drive interface cable. This interface is buffered and capable of driving a 10 foot cable. J4 is a 50 pin connector as is J3 the tape drive ribbon cable connector. The mating plug for both J3 and J4 is part no. 51-1150-11 (Winchester Products IDC socket connector with strain relief). Tables 3-4, 5-1, 5-2 and 6-1 specify the signal pin assignments for J1, J2, J3, and J4, respectively.

J5 is the power connector. It has ten contacts, pins 8 and 9 are for +5V and pins 1, 2, and 4 are GND, the rest are unused. The mating connector for J5 is part no. Molex 09-50-3101 with 08-50-0106 inserts.

2.3 MOUNTING

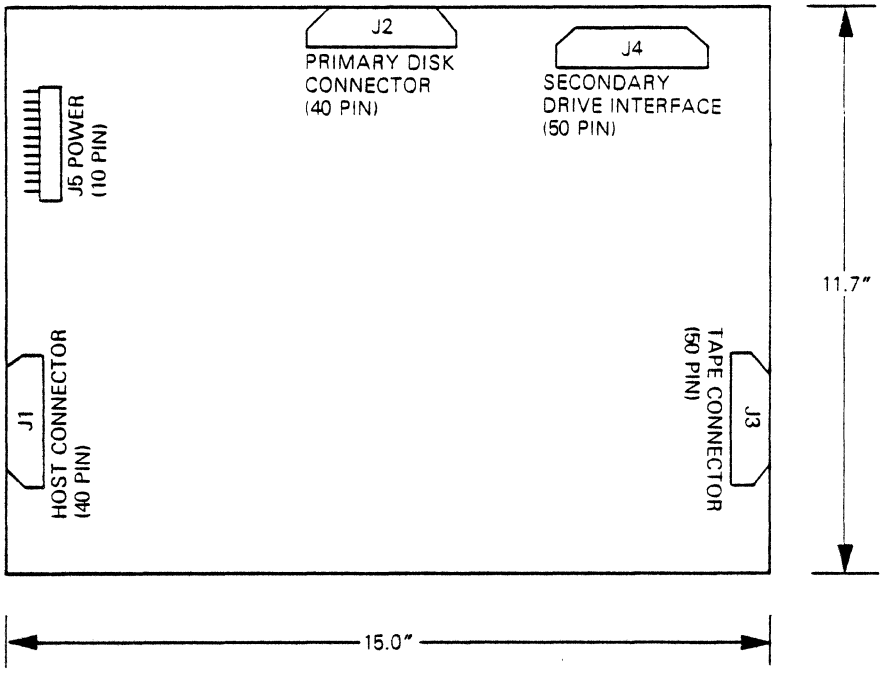
The T series board mounts on a Marksman drive in the area reserved for the disk interface device. It is secured to the drive via 16 "Snap On" retainers.

2.4 POWER CONSUMPTION

Power consumption is 3.2A max. at +5V for the formatter board only.

2.5 ENVIRONMENTAL LIMITS

The operating temperature limits are 32°F to 158°F (0°C to 70°C) with humidity from 10% to 90%, non-condensing.



NOTES: COMPONENT SIDE VIEW

Figure 2-1. Physical Configuration

SECTION 3
HOST INTERFACE

3.0 GENERAL

The formatter interface to the host is simple, both physically and architecturally. It can accommodate the complete range of host system capability, from a simple processor port access for each byte, to a full DMA approach, utilizing overlapped direct and transparent commands.

3.1 INTERFACE ARCHITECTURE

The host interfaces to the formatter via a 16 byte Command Register (CMDREG), a 16 byte Status Register (STREG), and a "high performance" discrete signal set. The host writes commands, command execution parameters, and data into the CMDREG. Note that the host cannot read back what it has written into the CMDREG. All read operations will read information from the STREG. Locations within the CMDREG are identified as CMDREG + X, where X is the (hexadecimal) offset from location 0. For instance, CMDREG + F is the FIFO location. Table 3-1 shows the definitions for each byte in the CMDREG.

Similarly, the STREG transfers status information from the formatter to the host. Locations within the STREG are identified as STREG + X, like the CMDREG. The host can always read information in the STREG when IMTBUSY is false, but cannot write to the STREG. All write operations will write to the CMDREG. Table 3-2 shows the definitions for each byte in the STREG. In a basic configuration, the host need use only the CMDREG and STREG to exercise the full formatter instruction set.

For high performance systems which require DMA and overlapped command capability, the formatter provides an additional six discrete status signals, which are the "high performance" signal set. These include two direct command status signals (CDCPT and IMTBUSY), two transparent command status signals (XPCPT and XPBUSY), and two DMA transfer signals (DTRQST/ and ACK/). With these six discrete signals, the host can monitor direct and transparent command status without reading the Status Byte (STREG + 0), it can generate interrupts upon command completion. and it can set up multiple sector DMA transfers which will carry through to completion without further host processor intervention. For further definition of the "high performance" signal set, see Section 3.2 — Interface Signal Description.

For host convenience, the STREG contains images of all the "high performance" discrete signals except ACK/ in STREG + 0, the formatter Status Byte. However, the user should exercise caution in using both the signals and their images in the same interface, because only IMTBUSY and DTRQST/ will change state simultaneously with their respective images. The CDCPT image in STREG + 0 will change state within 20 microseconds of the CDCPT signal, while the XPCTP and XPBUSY images may change state seconds after transitions on the corresponding signals. In all cases, a state change in the discrete signal will indicate the actual event, while the image in STREG + 0 will change only after the Host has accessed the formatter and it has updated STREG + 0.

3.1.1 Command Register

The host communicates command parameters, commands, and data, to the formatter through the 16 byte COMMAND REGISTER (CMDREG). To the host, the CMDREG appears as "write only" memory, while to the formatter it appears as ROM. The host loads disk command parameters into CMDREG + 1 to CMDREG + 8, and loads the tape command parameter byte into CMDREG + D and CMDREG + E. Once it has loaded all the necessary parameters, the host writes the command into CMDREG + 0. For write commands, the host loads data into CMDREG + F, which is the FIFO. Table 3-1 defines all the locations in the CMDREG.

3.1.2 Status Register

The host reads formatter status, command completion codes, and data from the STATUS REGISTER (STREG), which appears to the host as ROM. The formatter stores disk status information in STREG + 2 through STREG + 8 and tape status information in STREG + 9 through STREG + E. The host reads data from the FIFO at STREG + F. Table 3-2 defines the locations in the STREG.

The first byte in the STREG, the formatter status byte, contains the images of the high performance signal set and two error signals. Two of these signals, DTRQST/ (bit 7) and IMTBUSY (bit 5) are always valid, the other bytes in the STREG being valid only when IMTBUSY is false. Table 3-3 lists the definitions for each bit in the status byte.

TABLE 3-1. COMMAND REGISTER (CMDREG) DEFINITION

Location	Definition	Comments
CMDREG + 0	Command Byte	See Tables 4-1, 4-8, 4-12, and 4-14 for commands.
CMDREG + 1	Disk Drive Select No.	0 or 1.
CMDREG + 2 CMDREG + 3	Cylinder Address MSB Cylinder Address LSB	See Table A-1 for valid parameter limits.
CMDREG + 4	Head Select Number	See Table A-1 for valid parameter limits.
CMDREG + 5	Start Sector Address	See Table A-1 for valid parameter limits.
CMDREG + 6	User definable byte	
CMDREG + 7	No. of Retrys if disk error occurs.	
CMDREG + 8	Number of consecutive sectors, cylinders, blocks, or file marks.	For multiple sector, cylinder, block or file mark commands only.
CMDREG + 9	Transparent Command Priority	
CMDREG + A to CMDREG + C	Reserved	
CMDREG + D	Tape Command Parameters	
CMDREG + E	Tape Drive Select	0, 1, 2, or 3.
CMDREG + F	Data FIFO write location.	

TABLE 3-2. STATUS REGISTER (STREG) DEFINITION

Location	Definition	Comments
STREG + 0	Formatter Status Byte	See Table 3-3 for bit definition
STREG + 1	Command Completion Byte	See Tables 4-4, 4-11, and 4-13 for definitions of completion codes.
STREG + 2 STREG + 3 STREG + 4 STREG + 5 STREG + 6 STREG + 7	Sync Byte Head No. + Cyl. MSB Cyl. LSB Sector User Byte ID field checksum	Disk status bytes. Relative bytes +1 → +6 taken directly from ID field of last sector read by controller. (See Table B-1 for ID field expansion). Note that disk bytes are valid only after execution of a disk access command.
STREG + 8	Number of sectors, cylinders, blocks transferred, or file marks read.	For multiple sector, cylinder, or block access commands, or file mark advance.
STREG + 9 STREG + A STREG + B STREG + C STREG + D STREG + E	Tape Status Byte 0 Tape Status Byte 1 Tape Status Byte 2 Tape Status Byte 3 Tape Status Byte 4 Tape Status Byte 5	See Table 4-10 for definition of tape status words. Note that tape status bytes are valid only after execution of certain tape or disk/tape commands.
STREG + F	Data FIFO location	

TABLE 3-3. FORMATTER STATUS BYTE DEFINITION (STREG + 0)

Bit	Signal Name	Comments
7	DTRQST/	Image of discrete signal DTRQST/.
6	RESERVED	
5	IMTBUSY	Image of discrete signal IMTBUSY.
4	CDCPT	Delayed image of discrete signal CDCPT.
3	XPBUSY	Delayed image of discrete signal XPBUSY.
2	XPCPT	Delayed image of discrete signal XPCPT.
1	CDOVRUN	Command Overrun. Indicates that host has issued a command without acknowledging completion of previous command.*
0	XPOVRUN	Transparent command overrun. Indicates that host has issued a transparent command without acknowledging completion of previous transparent command.*

*See Section 4.7 for details of host command completion acknowledge.

STREG + 1 contains the completion code for the command just executed. The code 00 (HEX) indicates good execution, while other codes vary depending on the type of command executed. Table 4-4, 4-8, 4-10, and 4-12 list the completion codes for disk, tape, disk/tape and system commands respectively.

3.2 INTERFACE SIGNAL DESCRIPTION

The T series formatter interfaces to the host system via a 40-connector ribbon cable (see Section 2.0 for connector specifications). Table 3-4 defines the pin assignments for the host connector, J1.

The interface signal set includes an eight bit data bus, a four bit address bus, and other control status signals. The signal set supports both polled and interrupt driven operations, as well as DMA operations.

3.2.1 Signal Levels

The interface signal set utilizes both active low and active high signals, for clarity and conformance with common microcomputer signal conventions. A slash (/) following the signal name designates an active low signal.

All signals assume standard TTL logic levels. For active high signals, the levels are:

True, logic 1 (high level) = +2.4V to +5.0V

False, logic 0 (low level) = 0.0V to +0.8V

For active low signals, the voltage ranges remain the same, but the logical sense is reversed. Signal voltages are measured at the controller connector, over a maximum cable length of 15 feet (4.5 meters).

TABLE 3-4. HOST CONNECTOR PIN ASSIGNMENTS

Pin	Signal	Pin	Signal
1	GROUND	2	HDAT0
3	HDAT1	4	HDAT2
5	HDAT3	6	HDAT4
7	HDAT5	8	HDAT6
9	HDAT7	10	GROUND
11	HRD/	12	GROUND
13	HWR/	14	HADR3
15	HADR2	16	HADR1
17	HADR0	18	GROUND
19	HRESET/	20	GROUND
21	CDCPT	22	GROUND
23	HBDIR/	24	IMTBUSY
25	GROUND	26	DTRQST/
27	GROUND	28	XPBUSY
29	GROUND	30	XPCPT
31	GROUND	32	ACK/
33	8 MHZ	34	RESERVED
35	8 MHZ/	36	RESERVED
37	RESERVED	38	RESERVED
39	RESERVED	40	RESERVED

3.2.2 Signal Terminations

All signals are terminated with 220 ohms to +5V and 330 ohms to GND. Signals from the host to the formatter are terminated in the formatter, while signals from

the formatter to the host are terminated in the host. The bidirectional data bus is terminated both in the formatter and in the host.

3.2.3 Signal Loading

All signals from the formatter can drive one standard TTL load (1.6 mA) in addition to the terminator resistors. Likewise, all signals from the host should meet the same drive requirements.

3.2.4 Signal Definition

All interface signals fall into one of two sets – the CMDREG/STREG signal set, and the high performance signal set, as shown in Figure 3-1.

The CMDREG/STREG signal set contains all the signals required for a basic interface, including the data bus, address bus, and a line for each of the read, write, and reset strobes.

The high performance signal set includes four pairs of signals. The data exchange set, DTRQST/, HBDIR/, and ACK/, and the DMA clock pair, 8MHZ and 8MHZ/, are designed for use by a DMA controller. The command status pair, IMTBUSY and CDCPT, provides the capability for generating a system interrupt upon command completion and then clearing the interrupt upon acknowledgement. The transparent command status pair, XPBUSY and XPCPT, provides the same capability for transparent commands.

The specific definition for each interface signal is as follows.

DATA BUS (HDATO-7, 8 lines). The data bus communicates command/status bytes and data bytes between the controller and the host. Other signals in the interface define the type of data, direction, and destination.

ADDRESS BUS (HADRO-3, 4 lines). The host provides the desired command/status register address over the address bus. One of the registers (address F) is the top of the FIFO, so the host also uses the address bus to point to the FIFO for data transfers.

HOST READ/ (HRD/, 1 line). After the address lines are stable, the host asserts HRD/ to indicate a read of the addressed location. See Figure 3-2 for STREG Read Timing, and Figure 3-4 for FIFO Read Timing.

HOST WRITE/ (HWR/, 1 line). After the address lines are stable, the host asserts HWR/ to indicate a write to the addressed location. See Figure 3-3 for CMDREG Write Timing, and Figure 3-5 for FIFO Write Timing.

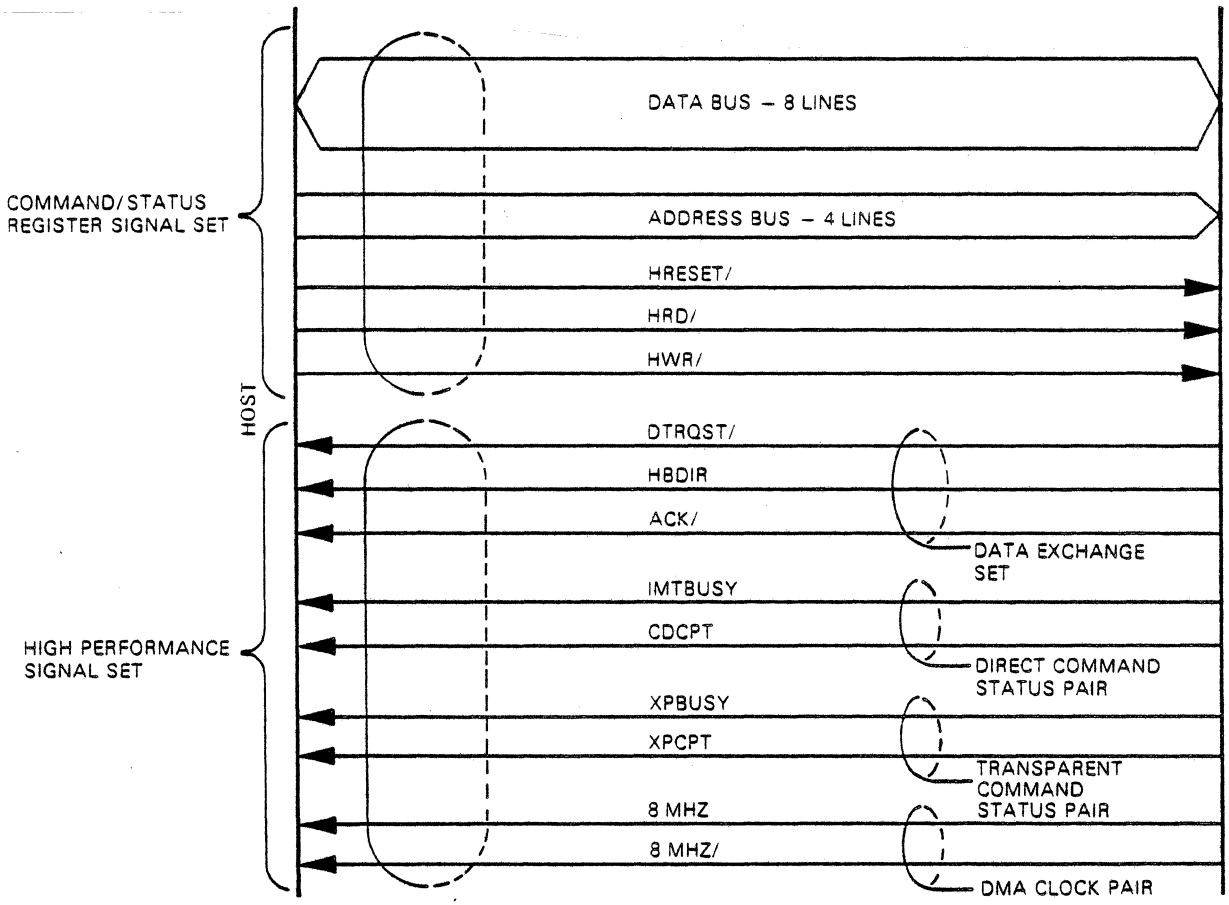


Figure 3-1. Interface Signal Set

HOST RESET/ (HRESET/, 1 line). Master reset for the formatter, disk drive, and tape drive.

DATA REQUEST/ (DTRQST/, 1 line). When active, indicates that the Host FIFO is ready for a data transfer. For a host read operation, DTRQST/ active indicates that the Host FIFO contains the requested data. For a host write operation, DTRQST/ active indicates the Host FIFO is ready to receive data from the host. For multiple sector operations, DTRQST/ will go false in between sector transfers, while the FIFO is under control of the formatter. Bit 7 of STREG+0 contains the image of DTRQST/.

TRANSFER ACKNOWLEDGE/ (ACK/, 1 line). The formatter uses ACK/ to acknowledge a byte transfer across the data bus. For a host write operation, the formatter asserts ACK/ after it has read the word from the data bus, and removes ACK/ after the host removes HWR/. For a host read operation, the formatter asserts ACK/ after the data is available on the data bus, and drops ACK/ after the host removes HRD/. See Figures 3-4 and 3-5 for the details of data read and write sequences.

HOST BUS DIRECTION (HBDIR, 1 line). When DTRQST/ is active, HBDIR indicates the direction of data flow across HDAT0 → HDAT7. A high level indicates data flow from the controller to the Host while a low level indicates flow from the Host to the formatter.

DMA CLOCK (8MHZ and 8MHZ/, 2 lines) 8MI

direct command. The host may access only the FIFO and bits 7 and 5 of STREG+0 while IMTBUSY is active. Bit 5 of STREG+0 contains the image of IMTBUSY.

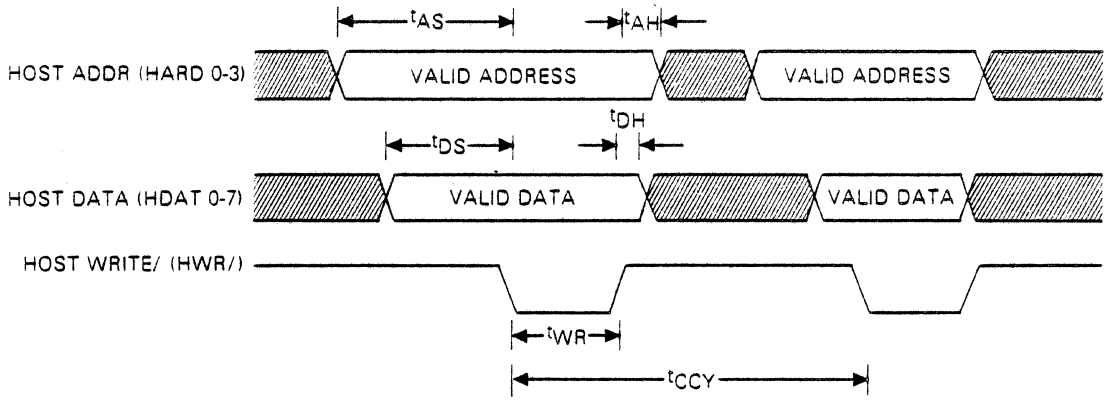
COMMAND COMPLETE (CDCPT, 1 line). When active, CDCPT indicates that the formatter has completed a command and is waiting for host acknowledgement, via the COMMAND ACKNOWLEDGE command. Bit 4 of STREG+0 contains the delayed image of CDCPT.

TRANSPARENT COMMAND BUSY (XPBUSY, 1 line). XPBUSY, together with XPCPT, indicates the status of transparent commands. When active, XPBUSY indicates that the formatter is currently executing a transparent command. However, XPBUSY has no affect on host access to the CMDREG/STREG, so the host is free to issue any other command (except another transparent command) while XPBUSY is active. Bit 3 of STREG+0 contains the image of XPBUSY.

TRANSPARENT COMMAND COMPLETE (XPCPT, 1 line). The formatter asserts XPCPT when it completes a transparent command. The host acknowledges XPCPT by issuing a READ TRANSPARENT COMMAND STATUS command. Bit 2 of STREG+0 contains the image of XPCPT.

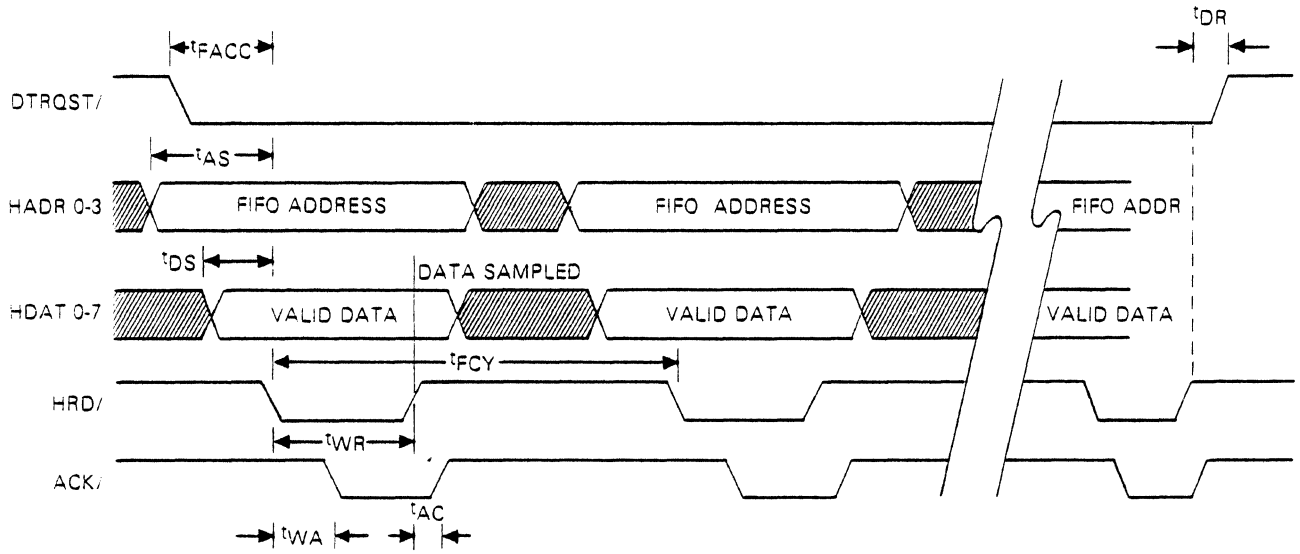
3.3 TIMING

The FIFO conforms to the same timing requirements as the CMDREG and STREG, thus easing the interface burden for systems in which the CPU accesses the FIFO. For DMA applications, the formatter provides an asynchronous handshake signal, ACK/, which signals the DMA unit when each data transfer is complete, thus allowing DMA transfers at the maximum data transfer rate. Figures 3-4 and 3-5 illustrate FIFO read and write timing.



t_{AS}	- ADDRESS STABLE TO HRD/ LOW	>30 ns
t_{DS}	- DATA STABLE TO HWR/ LOW	>30 ns
t_{AH}	- ADDRESS HOLD AFTER WRITE	>0 ns
t_{DH}	- DATA HOLD AFTER WRITE	>0 ns
t_{WR}	- WRITE STROBE DURATION	⩾250 ns
t_{CCY}	- CSR CYCLE TIME (HRW/ LOW TO NEXT HRW/ LOW)	⩾500 ns

Figure 3-3. Command Register Write Timing



t_{ACC}	- FIFO READY	>0 ns	(DTRQST/ LOW TO FIRST HWR/ LOW)
t_{AS}	- ADDRESS SETTLING TIME	>30 ns	(ADDRESS STABLE TO HWR/ LOW)
t_{DS}	- DATA SETTLING TIME	>30 ns	(DATA STABLE TO HWR/ LOW)
t_{WR}	- WRITE STROBE DURATION	≥250 ns	
t_{FCY}	- FIFO CYCLE TIME	500 ns	
t_{WA}	- ACK/ ASSERTION AFTER WRITE ASSERTION	<250 ns	
t_{AC}	- ACK/ CLEAR TIME AFTER WRITE REMOVED	<30 ns	
t_{DR}	- DTRQST/ REMOVAL AFTER FIFO EMPTY	<50 ns	

Figure 3-5. FIFO Write Timing

SECTION 4
COMMAND SET

4.0 GENERAL

The T series formatter command set minimizes the burden on the host system in interfacing to the disk and tape drives. It provides a full set of disk commands, a full set of tape commands, and four joint disk/tape commands for back-up and restore operations. In addition, to maximize disk availability to the host, the formatter has two command types, direct and transparent. The host may issue a direct command while a transparent command is executing, thus reducing delays in accessing the disk. See Section 4.3 — DIRECT and TRANSPARENT COMMANDS, for further discussion of command types.

4.1 COMMAND SEQUENCE (Single Sector Operation)

When the formatter is initialized and awaiting a command, IMTBUSY and DTRQST/ will be false. The host issues a command to the formatter by first writing the appropriate parameters into CMDREG+1 through CMDREG+E, then writing the command into CMDREG+0. The IMTBUSY signal is asserted immediately following a host write to CMDREG+0, and remains true until command execution is complete. The host may monitor IMTBUSY via the discrete signal or by reading bit 5 of STREG+0. When IMTBUSY is true, only bits 7 and 5 of STREG+0 are valid. All other bits in STREG+0 and all other bytes in the STREG except for the FIFO (at STREG+F) are invalid until IMTBUSY returns false, upon command completion.

Once the host has issued a command, DTRQST/ will be asserted, indicating that the host may access the FIFO. For write operations, the host will load data into the FIFO for subsequent storage onto the disk or tape. For read operations, the host will read requested data from the FIFO. One of the command parameters for read or write operations is the "number of consecutive sectors of data to transfer," which the host writes into CMDREG+8 prior to issuing the command. The formatter has an internal data counter which will not allow command termination until the designated number of bytes has been transferred; therefore, the host must transfer the correct number of bytes.

Once a sector of data has been transferred, DTRQST/ goes false while IMTBUSY remains true.

Upon command completion, the formatter removes IMTBUSY and asserts CDCPT. The host then reads the command completion code and status in the STREG and issues the COMMAND COMPLETION ACKNOWLEDGE (CDACK) command to clear the CDCPT signal. If the host issues another command before issuing the CDACK command, the Command Overrun Error, bit 1 in the STREG+0, will be set. The previous command completion status will be lost, and the formatter will execute the newly issued command. Completion status for the new command will be available in the STREG upon command completion.

4.2 MULTIPLE SECTOR ACCESS

The host indicates a multiple sector access command by setting CMDREG+8 to a number greater than one. The command sequence is the same for a single sector access except that DTRQST/ may be removed following transfer of the last byte in a sector, as shown in Figures 3-3 and 3-4, which detail FIFO read and write timing. The host may not access the FIFO until DTRQST/ is reasserted. Because DTRQST/ will go false, thus pausing the data transfer several times for a large data transfer, the host DMA controller may use it to initiate the DMA bus request and relinquish operations.

4.3 DIRECT AND TRANSPARENT COMMANDS

The T series formatter has two command types — direct and transparent. Heirarchically, direct commands have priority over transparent commands, which means the host may issue a direct command while a transparent command is executing. The formatter will interrupt the transparent command, execute the direct command, then finish executing the transparent command. The host may not issue a direct command while another direct command is in execution, but may interrupt any transparent command as often as necessary by issuing a direct command. Obviously, the host may not issue a transparent command while any other command, direct or transparent, is executing.

The purpose of having two command types is to maximize disk availability to the host. Direct command execution time, primarily determined by the disk seek time and rotational latency, and number of sectors accessed, is in the millisecond range. But transparent commands,

which involve tape positioning, may require tens of seconds to execute. In order to prevent the host from being "locked out" from disk access during the tape positioning time, most tape commands are transparent. Thus, the host can continue to access the disk while the tape is positioning for a disk/tape transfer.

The sequence for a transparent command is slightly different than for a direct command. The host writes command parameters in CMDREG+1 through CMDREG+E, then writes the command in CMDREG+0, just as in a direct command. However, to monitor transparent command completion, the host must use either the two discrete signals XPBUSY and XPCPT or the READ TRANSPARENT COMMAND STATUS (RXS) command. The signal pair XPBUSY and XPCPT has the same definitions for transparent commands that IMTBUSY and CDCPT have for direct commands.

Hosts which do not monitor the discrete transparent command status signals can determine command status via the RXS command. Bits 3 and 2 of STREG+0 reflect the status of XPBUSY and XPCPT, repsectively, following the RXS command. The ACKNOWLEDGE TRANSPARENT command is used to acknowledge completion of a transparent command. When XPCPT is true (upon command completion), issuance of the ACKXS command will reset XPCPT. Note that although XPBUSY and XPCPT are always present in STREG+0, their state is not current except immediately following issuance of a command. Therefore, if a transparent command is pending, the host must either look at the discrete signal XPBUSY or issue a command (preferably the RXS command) to determine if the transparent command is complete.

If the host does not issue the ACKXS command prior to issuing another transparent command, a Transparent Status Overrun error will result. Status of the previous transparent command is lost, but status of the current command will be available upon command completion. Bit 0 of STREG+0, the transparent command overrun error bit, is valid whenever the host issues the RXS command.

4.4 DISK COMMANDS

Table 4-1 below lists the disk commands, which together make up a powerful, high level disk interface. Table 4-2 details the CMDREG definition for disk commands and Table 4-3 does the same for the STREG. Table 4-4 shows the completion codes for disk commands. The remainder of this subsection describes each disk command in detail, including contents of the CMDREG at command initiation, contents of the STREG at command completion, and the operational sequence for the command.

TABLE 4-1. DISK COMMANDS

Command	Opcode (Hex)	Definition
BOOT	10	Read drive 0, head 0, cylinder 0, sector 0.
SEQUENCE DRIVE UP	20	
SEQUENCE DRIVE DOWN	30	
FORMAT	40 41 42 43	Format track with no verify. Format track with CRC verify. Format track with FIFO data fill no verify. Format track with FIFO data fill and CRC verify.
READ	58	Read with user specified retry count.
WRITE	60	Write.
CRC TRACK VERIFY	70	Verify a track with read and CRC.
READ DRIVE STATUS	80	Read drive status and current cylinder position.
SEEK	90	Command drive to seek target cylinder.
DRIVE IDENT.	A0	Read drive ID and sector size from unidentified drive.

TABLE 4-2. COMMAND REGISTER DEFINITION FOR DISK COMMANDS

Location	Definition	Comments
CMDREG + 0	Command Byte	See Tables 4-1 for Disk commands.
CMDREG + 1	Disk Drive Select No.	0 or 1.
CMDREG + 2 CMDREG + 3	Cylinder Address MSB Cylinder Address LSB	See Table A-1 for valid parameter limits.
CMGREG + 4	Head Select Number	See Table A-1 for valid parameter limits.
CMDREG + 5	Start Sector Address	See Table A-1 for valid parameter limits.
CMDREG + 6	User definable byte	
CMDREG + 7	# of read retries	
CMDREG + 8	Number of consecutive sectors to access.	For multiple sector access commands only.
CMDREG + 9	Transparent Command Priority	
CMDREG + A to CMDREG + E	Not used	
CMDREG + F	Data FIFO write location	

TABLE 4-3. STATUS REGISTER DEFINITION FOR DISK COMMANDS

Location	Definition	Comments
STREG + 0	Formatter Status Byte	See Table 3-3 for bit definition.
STREG + 1 STREG + 2 STREG + 3 STREG + 4 STREG + 5 STREG + 6 STREG + 7	Command Completion Byte Sync Byte Head No. + Cyl. MSB Cyl. LSB Sector User Byte ID field checksum	See Table 4-4 for definition Disk status bytes. Relative bytes +1 → +6 taken directly from ID field of last sector read by controller. (See Table B-1 for ID field expansion.) Note that disk bytes are valid only after execution of a disk access command.
STREG + 8	Number of sectors transferred	For multiple sector commands only.
STREG + 9 to STREG + E	Not used	
STREG + F	Data FIFO location	

TABLE 4-4. DISK COMMAND COMPLETION CODES

Code	Definition
00 (Hex)	Good execution
01	Good execution; seek retry after seek fault error
02	Good execution; seek retry after cylinder mismatch error
04	Good execution; data retry after ID checksum error
05	Good execution; data retry after CRC error
13	Drive Seeking
20	Disk Drive not present
21	Seek fault
22	Cylinder mismatch error
23	Sector not found
24	ID checksum error
25	Data CRC error
28	Head Write Protected
31	Invalid disk drive (other than 0 or 1)
32	Invalid head
33	Invalid cylinder
34	Invalid sector
36	Invalid drive ID
37	Invalid sector size (other than 128, 256, or 512 bytes)*
38	Disk (Cylinder Max) over run
40	Bad sector detected by CRC verify routine

*1024 byte sector can not be used.

4.4.1 BOOT Command (Direct)

INPUTS:

		Command Byte
CMDREG + 0	BIT	7 6 5 4 3 2 1 0
(10 Hex)	VALUE	0 0 0 1 0 0 0 0
CMDREG + 1 to CMDREG + E		Not used

OUTPUTS:

STREG + 0	Formatter status byte
STREG + 1	Formatter completion code
STREG + 2	Sync byte from ID field of last sector read
STREG + 3	Head + cylinder number MSB
STREG + 4	Cylinder number LSB
STREG + 5	Sector number
STREG + 6	User defined byte
STREG + 7	ID field checksum
STREG + 8 to STREG + E	Not used

BOOT sequence

1. Assert IMTBUSY.
2. Check for disk drive no. 0 on line and sequenced up.
   2.1 Issue sequence up command if necessary.
3. Read sector number 0 of drive number 0, cylinder number 0, head number 0.
   3.1 Issue seek if necessary to read sector.
4. Return sector data in FIFO.
5. Assert DTRQST/.
6. When host has read sector from FIFO, remove DTRQST/, update STREG, remove IMTBUSY, and assert CDCPT.

4.4.2 SEQUENCE DRIVE UP Command (Direct)

INPUTS:

		Command Byte
CMDREG + 0	BIT	7 6 5 4 3 2 1 0
(20 Hex)	VALUE	0 0 1 0 0 0 0 0
CMDREG + 1		Disk drive select number (0 or 1)
CMDREG + 2 to CMDREG + E		Not used

OUTPUTS:

STREG + 0	Formatter status byte
STREG + 1	Formatter completion code
STREG + 2 to STREG + E	Not used

SEQUENCE DRIVE UP sequence

1. Assert IMTBUSY.
2. Check for valid disk drive select number.
3. Check for disk drive on line.
4. Issue sequence up command.
5. Update STREG, remove IMBUSY and assert CDCPT.

4.4.3 SEQUENCE DRIVE DOWN Command (Direct)

INPUTS:

		Command Byte
CMDREG + 0	BIT	7 6 5 4 3 2 1 0
(30 Hex)	VALUE	0 0 1 1 0 0 0 0

CMDREG + 1	Disk drive select number (0 or 1)
CMDREG + 2 to CMDREG + E	Not used

OUTPUTS:

STREG + 0	Formatter status byte
STREG + 1	Formatter completion code
STREG + 2 to STREG + E	Not used

SEQUENCE DRIVE DOWN sequence

1. Assert IMTBUSY.
2. Check for valid disk drive select number.
3. Check for disk drive on line.
4. Issue sequence down command.
5. Update STREG, remove IMTBUSY and assert CDCPT.

4.4.4 FORMAT Commands (Direct)

4.4.4.1 FORMAT TRACK WITH NO VERIFY (direct)

INPUTS:

		Command Byte
CMDREG + 0	BIT	7 6 5 4 3 2 1 0
(40 Hex)	VALUE	0 1 0 0 0 0 0 0

CMDREG + 1	Drive select number (0 or 1)
CMDREG + 2	Cylinder number MSB
CMDREG + 3	Cylinder number LSB
CMDREG + 4	Head select number
CMDREG + 5	Data fill byte
CMDREG + 6	ID Field User specified byte
CMDREG + 7 to CMDREG + E	Not used

OUTPUTS:

STREG + 0	Formatter status byte
STREG + 1	Formatter completion code
STREG + 2	Sync byte from ID field of last sector written
STREG + 3	Head + cylinder number MSB
STREG + 4	Cylinder number LSB
STREG + 5	Sector number
STREG + 6	User defined byte
STREG + 7	ID field checksum
STREG + 8 to STREG + E	Not used

FORMAT TRACK WITH NO VERIFY sequence

1. Assert IMTBUSY.
2. Check for valid disk drive select number.
3. Check for drive on-line and up-to-speed.
   3.1 Issue squence up command if necessary.
   3.2 Wait for drive up-to-speed.
4. Check for valid head select number and cylinder select number.
   4.1 Issue seek to selected cylinder if necessary.
   4.2 Wait for seek complete.
5. Format ID fields of all sectors on track and fill data fields with user specified byte (CMDREG + 5).
6. Update STREG, remove IMTBUSY, and assert CDCPT.

4.4.4.2 FORMAT TRACK WITH CRC VERIFY Command · (Direct)

INPUTS:

		Command Byte
CMDREG + 0	BIT	7 6 5 4 3 2 1 0
(41 Hex)	VALUE	0 1 0 0 0 0 0 1

CMDREG + 1	Disk drive select number (0 or 1)
CMDREG + 2	Cylinder number MSB
CMDREG + 3	Cylinder number LSB
CMDREG + 4	Head select number
CMDREG + 5	Data fill byte
CMDREG + 6	ID Field User specified byte
CMDREG + 7 to CMDREG + E	Not used

OUTPUTS:

STREG + 0	Formatter status byte
STREG + 1	Formatter completion code
STREG + 2	Sync byte from ID field of last sector read
STREG + 3	Head + cylinder number MSB
STREG + 4	Cylinder number LSB
STREG + 5	Sector number
STREG + 6	User defined byte
STREG + 7	ID field checksum

STREG + 8	Number of bad sectors detected
STREG + 9	
to	Not used
STREG + E	

FORMAT TRACK WITH CRC VERIFY sequence

1. Assert IMTBUSY.
2. Check for valid disk drive select number.
3. Check for drive on-line and up-to-speed.
   3.1 Issue sequence up command if necessary.
   3.2 Wait for drive up-to-speed.
4. Check for valid head select number and cylinder select number.
   4.1 Issue seek to selected cylinder if necessary.
   4.2 Wait for seek complete.
5. Format ID fields of all sectors on track and fill data fields with user specified byte (CMDREG + 5).
6. Perform a CRC READ on each sector in the track.
   6.1 Check ID field checksum.
   6.2 Check data CRC code.
   6.3 Indicate bad sector number in FIFO. The first location in FIFO will contain the number of the first bad sector, the second location in FIFO will contain the number of the second bad sector, etc.
   6.4 Indicate the number of bad sectors found in STREG + 8.
7. Assert DTRQST/.
8. When host has read one sector of data from FIFO remove DTRQST.
9. Remove IMTBUSY and assert CDCPT.

4.4.4.3 FORMAT TRACK WITH FIFO DATA AND NO VERIFY Command (Direct)

INPUTS:

		Command Byte
	BIT	7 6 5 4 3 2 1 0
CMDREG + 0 (42 Hex)	VALUE	0 1 0 0 0 0 1 0

CMDREG + 1	Drive select number (0 or 1)
CMDREG + 2	Cylinder number MSB
CMDREG + 3	Cylinder number LSB
CMDREG + 4	Head select number
CMDREG + 5	Not used
CMDREG + 6	ID Field User specified byte
CMDREG + 7	
to	Not used
CMDREG + E	

OUTPUTS:

STREG + 0	Formatter status byte
STREG + 1	Formatter completion code
STREG + 2	Sync byte from ID field of last sector written
STREG + 3	Head + cylinder number MSB
STREG + 4	Cylinder number LSB
STREG + 5	Sector number
STREG + 6	User defined byte
STREG + 7	ID field checksum
STREG + 8	
to	Not used
STREG + E	

FORMAT TRACK WITH NO VERIFY sequence

1. Assert IMTBUSY.
2. Check for valid disk drive select number.
3. Check for drive on-line and up-to-speed.
   3.1 Issue squence up command if necessary.
   3.2 Wait for drive up-to-speed.
4. Check for valid head select number and cylinder select number.
   4.1 Issue seek to selected cylinder if necessary.
   4.2 Wait for seek complete.
5. Assert DTRQST/.
6. After host transfers sector of data containing fill pattern drop DTRQST/.
7. Format ID fields of all sectors on track and fill data fields with data in FIFO.
8. Update STREG, remove IMTBUSY, and assert CDCPT.

4.4.4.4 FORMAT TRACK WITH FIFO DATA AND CRC VERIFY Command (Direct)

INPUTS:

		Command Byte
	BIT	7 6 5 4 3 2 1 0
CMDREG + 0 (43 Hex)	VALUE	0 1 0 0 0 0 1 1

CMDREG + 1	Disk drive select number (0 or 1)
CMDREG + 2	Cylinder number MSB
CMDREG + 3	Cylinder number LSB
CMDREG + 4	Head select number
CMDREG + 5	Not used
CMDREG + 6	ID Field User specified byte
CMDREG + 7	
to	Not used
CMDREG + E	

OUTPUTS:

STREG + 0	Formatter status byte
STREG + 1	Formatter completion code
STREG + 2	Sync byte from ID field of last sector read
STREG + 3	Head + cylinder number MSB
STREG + 4	Cylinder number LSB
STREG + 5	Sector number
STREG + 6	User defined byte
STREG + 7	ID field checksum
STREG + 8	Number of bad sectors detected
STREG + 9	
to	Not used
STREG + E	

FORMAT TRACK WITH CRC VERIFY sequence

1. Assert IMTBUSY.
2. Check for valid disk drive select number.
3. Check for drive on-line and up-to-speed.
   3.1 Issue sequence up command if necessary.
   3.2 Wait for drive up-to-speed.
4. Check for valid head select number and cylinder select number.
   4.1 Issue seek to selected cylinder if necessary.
   4.2 Wait for seek complete.
5. Assert DTRQST/.
6. After Host has transferred sector containing fill pattern drop DTRQST/.
7. Format ID fields of all sectors on track and fill data fields with FIFO data.
8. Perform a CRC READ on each sector in the track.
   8.1 Check ID field checksum.
   8.2 Check data CRC code.
   8.3 Indicate bad sector number in FIFO. The first location in FIFO will contain the number of the first bad sector, etc.
   8.4 Indicate the number of bad sectors found in STREG + 8.
9. Assert DTRQST/.
10. When host has read sector of data from FIFO, remove DTRQST/, remove IMTBUSY and assert CDCPT.

4.4.5 READ Command

4.4.5.1 READ SECTOR WITH RETRY Command (Direct)

INPUTS:

		Command Byte							
CMDREG + 0	BIT	7	6	5	4	3	2	1	0
(58 Hex)	VALUE	0	1	0	1	1	0	0	0

CMDREG + 1	Disk drive select number (0 or 1)
CMDREG + 2	Cylinder number MSB
CMDREG + 3	Cylinder number LSB
CMDREG + 4	Head select number
CMDREG + 5	Start sector number
CMDREG + 6	Not used
CMDREG + 7	Number of read retries if error detected
CMDREG + 8	Number of consecutive sectors to read (0 defaults to 1)
CMDREG + 9 to CMDREG + E	Not used

OUTPUTS:

STREG + 0	Formatter status byte
STREG + 1	Formatter completion code
STREG + 2	Sync byte from ID field of last sector read
STREG + 3	Head + cylinder number MSB
STREG + 4	Cylinder number LSB
STREG + 5	Sector number
STREG + 6	User defined byte
STREG + 7	ID field checksum
STREG + 8	Number of sectors read
STREG + 9 to STREG + E	Not used

READ SECTOR WITH RETRY sequence

1. Assert IMTBUSY.
2. Check for valid disk drive select number.
3. Check for drive on-line and up-to-speed.
   3.1 Issue sequence up command if necessary.
   3.2 Wait for drive up-to-speed.
4. Check for valid head, cylinder, and sector select numbers.
   4.1 Issue seek to selected cylinder if necessary.
   4.2 Wait for seek complete.
5. Search track for sector preceding selected sector.
6. Read data from selected sector and store in FIFO.
7. Check ID field checksum and data CRC code.
8. If error is detected, repeat steps 5 through 7 up to (CMDREG + 7) times.
   8.1 If after all rereads, data is still in error, abort command. Update STREG, remove IMTBUSY and assert CDCPT.
9. Assert DTRQST/.
10. After host has read full FIFO block, or ending partial FIFO raise DTRQST/.
11. Rotate FIFOs.
12. Repeat steps 5 through 11 until host has read data from all requested sectors.
13. Remove and IMTBUSY, and assert CDCPT.

Note

For Programmed I/O Block Move, the host should check DTRQST after each sector transfer.

4.4.6 WRITE Command

4.4.6.1 WRITE SECTOR Command (Direct)

INPUTS:

		Command Byte							
CMDREG + 0	BIT	7	6	5	4	3	2	1	0
(60 Hex)	VALUE	0	1	1	0	0	0	0	0

CMDREG + 1	Disk select number (0 or 1)
CMDREG + 2	Cylinder number MSB
CMDREG + 3	Cylinder number LSB
CMDREG + 4	Head select number
CMDREG + 5	Start sector number
CMDREG + 6	ID Field User specified byte
CMDREG + 7	Not used
CMDREG + 8	Number of consecutive sectors to write
CMDREG + 9 to CMDREG + E	Not used

OUTPUTS:

STREG + 0	Formatter status byte
STREG + 1	Formatter completion code
STREG + 2	Sync byte from ID field of last sector written
STREG + 3	Head + cylinder number MSB
STREG + 4	Cylinder number LSB
STREG + 5	Sector number
STREG + 6	User defined byte
STREG + 7	ID field checksum
STREG + 8	Number of sectors written
STREG + 9 to STREG + E	Not used

WRITE SECTOR sequence

1. Assert IMTBUSY.
2. Check for valid disk drive select number.
3. Check for drive on-line and up-to-speed.
   3.1 Issue sequence up command if necessary.
   3.2 Wait for drive up-to-speed.
4. Check for valid head, cylinder, and sector select numbers.
   4.1 Issue seek to selected cylinder if necessary.
   4.2 Wait for seek complete.
5. Assert DTRQST/.
6. After host has transferred complete FIFO or last partial FIFO remove DTRQST/.
7. Search track for sector preceding selected sector.
8. Write data from FIFO to selected sectors.
9. Repeat steps 5 through 8 until host has written data to all sectors.
10. Remove IMTBUSY, and assert CDCPT.

Note

For Programmed I/O Block move data transfers, the host should check DTRQST/ after each sector transfer.

4.4.7 CRC TRACK VERIFY Command (Direct)

INPUTS:

		Command Byte							
		7	6	5	4	3	2	1	0
CMDREG + 0 (70 Hex)	BIT								
	VALUE	0	1	1	1	0	0	0	0

CMDREG + 1	Disk select number (0 or 1)
CMDREG + 2	Cylinder number MSB
CMDREG + 3	Cylinder number LSB
CMDREG + 4	Head select number
CMDREG + 5 to CMDREG + E	Not used

OUTPUTS:

STREG + 0	Formatter status byte
STREG + 1	Formatter completion code
STREG + 2	Sync byte from ID field of last sector read
STREG + 3	Head + cylinder number MSB
STREG + 4	Cylinder number LSB
STREG + 5	Sector number
STREG + 6	User defined byte
STREG + 7	ID field checksum
STREG + 8	Number of bad sectors
STREG + 9 to STREG + E	Not used

CRC TRACK VERIFY sequence

1. Assert IMTBUSY.
2. Check for valid disk drive select number.
3. Check for drive on-line and up-to-speed.
   3.1 Issue sequence up command if necessary.
   3.2 Wait for drive up-to-speed.
4. Check for valid head and cylinder select numbers.
   4.1 Issue seek to selected cylinder if necessary.
   4.2 Wait for seek complete.
5. Perform a CRC read on each sector in track.
   5.1 Check ID field checksum.
   5.2 Check data CRC code.
6. Indicate bad sector number in FIFO. First byte in FIFO will contain the number of the first bad sector, second byte in FIFO will contain the number of the second bad sector, etc.
7. Store total number of bad sectors detected in STREG + 8.
8. Assert DTRQST/.
9. When host has read one sector from FIFO, remove DTRQST.
10. Remove IMTBUSY, then assert CDCPT.

4.4.8 READ DRIVE STATUS Command

INPUTS:

		Command Byte							
		7	6	5	4	3	2	1	0
CMDREG + 0 (80 Hex)	BIT								
	VALUE	1	0	0	0	0	0	0	0

CMDREG + 1	Disk drive select number (0 or 1)
CMDREG + 2 to CMDREG + E	Not used

OUTPUTS:

STREG + 0	Formatter status byte
STREG + 1	Formatter completion code
STREG + 2	Disk status register (see Table 4-5)
STREG + 3	Disk extended status register (see Table 4-5)
STREG + 4	Current cylinder number MSB
STREG + 5	Current cylinder number LSB
STREG + 6 to STREG + E	Not used

READ DRIVE STATUS Sequence

1. Assert IMTBUSY.
2. Check for valid disk drive select number.
3. Check for drive on-line.
4. Read disk status registers.
5. Read current cylinder position registers.
6. Update STREG, remove IMTBUSY, then assert CDCPT.

TABLE 4-5. MARKSMAN STATUS BYTES

Status Byte 0	
Bit	**Definition**
0	Track Zero: Heads located over track zero.
1	Landing Zone: Heads located over landing zone.
2	Illegal Command
3	Ready: Disk is up to rotational speed.
4	Spin acceleration/deceleration out of limit (M20/M40). Hardware Error (M80/M160)
5	End of Cylinder
6	Diagnostic error
7	Track zero error (M20/M40). Unsolicited Interrupt (M80/M160)
Status Byte 1 (Expanded Status Byte)	
Bit	**Definition**
0	Sector length switch (MSB)
1	Sector length switch (LSB)
2	Illegal Set Sector
3	Sector/track set by 0 = Sector Length Switches 1 = Set Sector Command
4	Illegal Rezero or Illegal Seek
5	Illegal Cylinder
6	Illegal Command
7	Attempted Write on Write Protect Head

4.4.9 SEEK Command (Direct)

INPUTS:

		Command Byte
CMDREG + 0	BIT	7 6 5 4 3 2 1 0
(90 Hex)	VALUE	1 0 0 1 0 0 0 0

CMDREG + 1 — Disk drive select number (0 or 1)
CMDREG + 2 — Cylinder select number MSB
CMDREG + 3 — Cylinder select number LSB
CMDREG + 4 to CMDREG + E — Not used

OUTPUTS:

STREG + 0 — Formatter status byte
STREG + 1 — Formatter completion code
STREG + 2 to STREG + E — Not used

SEEK sequence

1. Assert IMTBUSY.
2. Check for valid disk drive select number.
3. Check for drive on-line and valid cylinder select number.
   3.1 Issue seek to selected cylinder if necessary.
   3.2 Wait for seek completion.
4. Update STREG, remove IMTBUSY, then assert CDCPT.

4.4.10 DRIVE IDENTIFICATION Command

INPUTS:

		Command Byte
CMDREG + 0	BIT	7 6 5 4 3 2 1 0
(A0 Hex)	VALUE	1 0 1 0 0 0 0 0

CMDREG + 1 — Drive select number (0 or 1)
CMDREG + 2 to CMDREG + E — Not used

OUTPUTS:

STREG + 0 — Formatter status byte
STREG + 1 — Formatter completion code
STREG + 2 — Disk drive identification code (see Table 4-6)
STREG + 3 — Sector size MSB (see Table 4-7)
STREG + 4 — Sector size LSB (see Table 4-7)
STREG + 5 — Defect map status
  00 (Hex) = Defect map cannot be read or is disabled
  01 (Hex) = Defect map enabled and operating
STREG + 6 — Max. Cylinder Number (MSB)
STREG + 7 — Max. Cylinder Number (LSB)
STREG + 8 — Max. Head Number
STREG + 9 — Max. Sector Number
STREG + A to STREG + E — Not used

DRIVE IDENTIFICATION sequence

1. Assert IMTBUSY.
2. Check for valid disk drive select number.
3. Check for drive on-line.
4. Check on-board lookup table for drive type and sector size from drive. After request issue drive RESTORE command.
5. Update STREG, remove IMTBUSY, then assert CDCPT.

TABLE 4-6. MARKSMAN DRIVE ID CODES

Drive Model	ID Code (Hex)
M20	01
M40	02
M80	03
M160	04

TABLE 4-7. SECTOR SIZE CODES

Sector Size	Sector Size Code (Hex)	
	MSB	LSB
128 bytes	00	80
256 bytes	01	00
512 bytes	02	00

4.5 TAPE COMMANDS

The T series formatter has a complete set of tape instructions with which the user can control up to four tape drives. Table 4-8 lists the various tape commands in the formatter's repertoire.

The host uses CMDREG + 8, CMDREG + D, and CMDREG + E to specify command parameters for tape commands. It writes the tape command into CMDREG + 0 and writes data to the FIFO at CMDREG + F. Table 4-9 details the CMDREG assignments for tape commands.

TABLE 4-8. TAPE COMMANDS

Command	Opcode (Hex)	Definition
POSITION*	F0	Position the tape (Rewind) (Erase).
WRITE DATA	F1	Write data directly from the host to the tape.
WRITE FILE MARK	F5	Write a file mark.
READ DATA	F6	Read data directly from the tape to the host.
ADVANCE FILE MARK(S)*	FA	Advance tape to next file mark(s).
READ STATUS	FB	Read tape status.

*Indicates a transparent command.

TABLE 4-9. COMMAND REGISTER DEFINITION FOR TAPE COMMANDS

Location	Definition	Comments
CMDREG + 0	Command	See Table 4-5 for commands.
CMDREG + 1 to CMDREG + 7	Not used	
CMDREG + 8	Number of data blocks to transfer, or number of file marks to read.	For read or write commands, and advance file mark command.
CMDREG + 9 to CMDREG + C	Not used	
CMDREG + D	Tape command parameters	
CMDREG + E	Tape drive select	0, 1, 2, or 3
CMDREG + F	Data FIFO	

The formatter reports tape command status in STREG+0, STREG+1, and STREG+8 while STREG+9 through STREG+E report status for the tape drive itself. The tape status bytes are identical to the six status bytes that the Archive and Cipher tape drive sends in response to a READ STATUS command. Table 4-10 defines the STREG usage for Archive tape commands. Table 4-11 defines the STREG usage for DEI tape commands. STREG+E includes the eight tape status signals returned by the DEI drive.

The formatter reports the tape command completion code in STREG+1. For good execution the completion code is 00 Hex. Table 4-12 lists other complete codes and the corresponding error conditions.

4.5.0.1 Tape Read/Write Modes

The formatter internally maintains two flags which reflect the mode of the selected tape drive. The READ MODE is set when the last tape operation command was "read data". The WRITE Flag is set when the last tape operation command was "write data". In order to select a new tape drive both modes must be reset. The internal use of these modes allows the user to concatenate tape read or write operations by issuing successive commands to the formatter. Table 4-13 shows the logic used to set and reset each flag, while Table 4-14 shows the command which may be issued with each mode set.

4.5.0.2. Tape Availability

Whenever a tape is being utilized in a transparent command, the tape controller is being used to oversee the command, therefore, the host cannot utilize the same or another tape drive while a transparent operation is in progress. Any attempt by the host to issue a direct command utilizing any tape drive while a transparent command is in progress will result in a 73 Hex completion code.

TABLE 4-10. STATUS DEFINITION FOR ARCHIVE TAPE COMMANDS

Location	Definition	Comments
STREG + 0	Formatter status byte	See Table 3-3 for bit definition.
STREG + 1	Command completion code	See Table 4-12 for definition.
STREG + 2 to STREG + 7	Not used	
STREG + 8	Number of data blocks transferred.	For multiple block read or write commands only.
STREG + 9	Exception Byte 0	
Bit 7	Exception Flag	Set if any bit in STREG+9 is set.
Bit 6	Cartridge not in place	
Bit 5	Drive not online	
Bit 4	Tape Write protected	
Bit 3	End of tape	
Bit 2	Hard data error	
Bit 1	Block in error not found	
Bit 0	File mark detected	
STREG + A	Exception Byte 1	
Bit 7	Exception Flag	Set if any bit in STREG+A is set
Bit 6	Illegal command	
Bit 5	No data detected	
Bit 4	Unused	
Bit 3	Beginning of Tape	
Bit 2	Unused	
Bit 1	Unused	
Bit 0	Tape Reset has occurred	
STREG + B	Data Error Counter MSB	Number of blocks rewritten (Write op)
STREG + C	Data Error Counter LSB	Number of soft read errors (Read op)
STREG + D	Overrun Counter MSB	Number of extended gaps (Write op)
STREG + E	Overrun Counter LSB	Number of read buffer underruns (Read op)
STREG + F	Data FIFO read location	

TABLE 4-11. STATUS REGISTER DEFINITION FOR DEI TAPE COMMANDS

Location	Definition	Comments
STREG + 0	Formatter status byte	See Table 3-3 for bit definition.
STREG + 1	Command completion code	See Table 4-12 for definition.
STREG + 2 to STREG + 7	Not used	
STREG + 8	Number of data blocks transferred.	For multiple block read or write commands only.
STREG + 9 to STREG + D	Not used	
STREG + E	Tape Status Byte	
Bit 7	DIE/	Unrecoverable data error
Bit 6	URY/	Tape Unit Ready
Bit 5	EOM/	End of Memory (for tape write opn). End of Data (for tape read opn)
Bit 4	ROM/	Cartridge write protected
Bit 3	EOD/	End of Recorded Data
Bit 2	ERF/	Error Flag. Tape drive is correcting erroneous data from tape.
Bit 1	TMD/	Tape Mark Detected
Bit 0	SLD/	Tape Drive Unit Selected
STREG + F	Data FIFO read location	

TABLE 4-12. TAPE COMMAND COMPLETION CODES

Code	Definition
00 (Hex)	Good execution
08	Tape Read Operation Complete Terminated by File Mark
60	Tape drive not present
61	Tape cartridge write protected
62	File Mark Detected
63	Block in error not found
65	Hard Data error
68	Cartridge not in place
69	End of Tape
6A	Tape not online
6C	Bottom of Tape
6D	No data detected
6E	Illegal Command (to tape drive from formatter)
70	Invalid Command with Tape Write Mode set
71	Invalid Command with Tape Read Mode set
73	Tape not available (currently being used in transparent command)

4.5.1 POSITION Command (Transparent)

INPUTS:

		Command Byte
CMDREG + 0	BIT	7 6 5 4 3 2 1 0
(F0 Hex)	VALUE	1 1 1 1 0 0 0 0
CMDREG + 1 to CMDREG + C	Not used	
CMDREG + D	Operation Parameter 04 = Retension tape 02 = Erase tape 01 = Rewind tape	
CMDREG + E	Tape Drive Select (0 to 3 Hex)	

OUTPUTS:

STREG + 0	Formatter status byte
STREG + 1	Formatter completion code
STREG + 2 to STREG + 8	Not used
STREG + 9 to STREG + E	Tape status bytes (see Tables 4-10 and 4-11)

TABLE 4-13. TAPE READ/WRITE MODE CONTROL LOGIC

Mode	Set By	Reset By
TAPE READ	a. Tape Read Data Command b. Burst Restore Command c. Transparent Restore Command	a. Reading File Mark b. Rewind Command to Selected Drive c. Tape drive error
TAPE WRITE	a. Tape Write Data Command b. Burst Backup Command c. Transparent Backup Command	a. Writing File Mark b. Rewind Command to Selected Tape (will cause a file mark write if not already written) c. Tape drive error

TABLE 4-14. TAPE MODE/COMMAND PROTOCOL

Mode	Valid Tape Related Instructions When Mode Set
TAPE READ	a. Tape Read Data b. Burst Restore c. Transparent Restore d. POSITION (rewind) command (all other tape related instructions will return with a 71 completion code)
TAPE WRITE	a. Tape Write Data b. Write File Mark c. Burst Backup d. Transparent Backup e. POSITION (rewind) command (all other tape related instructions will return with a 70 completion code)

POSITION sequence

1. Remove IMTBUSY and assert XPBUSY.
2. Decode CMDREG + E.
3. Check for valid tape drive select number.
   3.1 Select tape drive.
4. Check for direct command.
   4.1 If direct command issued, execute it.
5. Check for tape position command completed.
6. Repeat steps 4 and 5 until tape is positioned.
7. Read tape status and update XP status buffer.
8. Remove XPBUSY and assert XPCPT.

4.5.2 WRITE DATA Command (Direct)

INPUTS:

		Command Byte
CMDREG + 0	BIT	7 6 5 4 3 2 1 0
(F1 Hex)	VALUE	1 1 1 1 0 0 0 1

CMDREG + 1 to CMDREG + 7	Not used
CMDREG + 8	Number of data blocks to transfer
CMDREG + 9 to CMDREG + D	Not used
CMDREG + E	Tape Drive Select (0 to 3 Hex)

OUTPUTS:

STREG + 0	Formatter status byte
STREG + 1	Formatter completion code
STREG + 2 to STREG + 7	Not used
STREG + 8	Number of data blocks transferred
STREG + 9 to STREG + E	Not used *

* Tape Status will not be returned after Write Data Command

WRITE DATA sequence

1. Assert IMTBUSY.
2. Check for valid tape drive select number.

3. Set Tape Mode to WRITE.
4. Assert DTRQST/.
5. Wait for host to write data to FIFO.
6. Remove DTRQST/.
7. Swap Host Buffer.
8. Assert DTRQST/.
9. Write data to tape.
10. Repeat steps 5 through 9 until host has written specified number of data blocks.
11. Update STREG, remove IMTBUSY, assert CDCPT.

4.5.3 WRITE FILE MARK Command (Direct)

INPUTS:

		Command Byte
CMDREG + 0	BIT	7 6 5 4 3 2 1 0
(F5 Hex)	VALUE	1 1 1 1 0 1 0 1

CMDREG + 1 to CMDREG + D	Not used
CMDREG + E	Tape Drive # (0 to 3 Hex)

OUTPUTS:

STREG + 0	Formatter status byte
STREG + 1	Formatter completion code
STREG + 2 to STREG + 8	Not used
STREG + 9 to STREG + E	Tape status bytes (see Tables 4-10 and 4-11)

WRITE FILE MARK sequence

1. Remove IMTBUSY and assert XPBUSY.
2. Write file mark.
3. Read tape drive status and store in XP status buffer.
4. Remove XPBUSY and assert CDCPT.

4.5.4 READ DATA Command (Direct)

INPUTS:

		Command Byte
CMDREG + 0	BIT	7 6 5 4 3 2 1 0
(F6 Hex)	VALUE	1 1 1 1 0 1 1 0

CMDREG + 1 to CMDREG + 7	Not used
CMDREG + 8	Number of data blocks to read
CMDREG + 9 to CMDREG + D	Not used
CMDREG + E	Tape Drive Select (0 to 3 Hex)

OUTPUTS:

STREG + 0	Formatter status byte
STREG + 1	Formatter completion code
STREG + 2 to STREG + 7	Not used
STREG + 8	Number of data blocks read
STREG + 9 to STREG + E	Not used *

*Tape drive status cannot be read during a TAPE read command until a file mark is encountered terminating the tape READ MODE.

READ DATA sequence

1. Assert IMTBUSY.
2. Check for valid tape drive select number.
3. Read data from tape.
4. Assert DTRQST/.
5. Wait for host to read 1 block of data from FIFO.
6. Remove DTRQST/.
7. Swap Host FIFO.
8. Repeat steps 5 thru 7 until host has read specified number of data blocks.
9. Update STREG, remove IMTBUSY, assert CDCPT.

4.5.5 ADVANCE FILE MARK(S) Command (Transparent)

INPUTS:

		Command Byte
CMDREG + 0	BIT	7 6 5 4 3 2 1 0
(FA Hex)	VALUE	1 1 1 1 1 0 1 0

CMDREG + 1 to CMDREG + 7	Not used
CMDREG + 8	Number of File Marks to Advance
CMDREG + 9 to CMDREG + D	Not used
CMDREG + E	Tape Drive Select (0 to 3 Hex)

OUTPUTS:

STREG + 0	Formatter status byte
STREG + 1	Formatter completion code
STREG + 2 to STREG + 8	Not used
STREG + 9 to STREG + E	Tape status bytes (see Tables 4-10 and 4-11)

ADVANCE FILE MARK(S) sequence

1. Remove IMTBUSY and assert XPBUSY.
2. Check for valid tape drive select number.
3. Advance one file mark.

4. Check for direct command.
   4.1 If direct command issued, execute it.
5. Repeat steps 3 - 4 until tape is positioned.
6. Read tape status and update XP status buffer.
7. Remove XPBUSY and assert XPCPT.

4.5.6 READ TAPE STATUS Command (Direct)

INPUTS:

		Command Byte
CMDREG + 0	BIT	7 6 5 4 3 2 1 0
(FB Hex)	VALUE	1 1 1 1 1 0 1 1

CMDREG + 1 to CMDREG + D	Not used
CMDREG + E	Tape Drive Select (0 to 3 Hex)

OUTPUTS:

STREG + 0	Formatter status byte
STREG + 1	Formatter completion code
STREG + 2 to STREG + 8	Not used
STREG + 9 to STREG + E	Tape status bytes (see Tables 4-10 and 4-11)

READ TAPE STATUS sequence

1. Remove IMTBUSY and assert XPBUSY.
2. Check for valid tape drive select number.
3. Read one byte of tape drive status and store in XP status buffer.
4. Remove XPBUSY and assert CDCPT.

4.6 DISK/TAPE COMMANDS

The T series formatter has four disk/tape transfer instructions which allow the host to command a disk/tape transfer. With the DISK BACK-UP commands, the host can command a transfer of up to 20 Mbytes from disk to tape which the formatter will execute without further host intervention. Similarly, the DISK RESTORE commands allow the host to perform the reverse operation. The transparent versions of each command are identical to the direct ones, except that they allow the host continued access to the disk. Table 4-15 lists the disk/tape commands, while Tables 3-1 and 3-2 define the CMDREG and STREG usage for these commands. Table 4-16 lists the completion codes and error conditions for disk/tape commands.

4.6.1 BURST DISK BACK-UP Command (Direct)

INPUTS:

		Command Byte
CMDREG + 0	BIT	7 6 5 4 3 2 1 0
(B0 Hex)	VALUE	1 0 1 1 0 0 0 0

CMDREG + 1	Drive select number (0 or 1)
CMDREG + 2	Cylinder number MSB
CMDREG + 3	Cylinder number LSB
CMDREG + 4 to CMDREG + 6	Not used
CMDREG + 7	Number of read retries if disk error occurs
CMDREG + 8	Number of consecutive cylinders to read
CMDREG + 9 to CMDREG + D	Not used
CMDREG + E	Tape Drive Select (0 to 3 Hex)

OUTPUTS:

STREG + 0	Formatter status byte
STREG + 1	Formatter completion code
STREG + 2	Sync byte from ID field of last sector read
STREG + 3	Head + cylinder number MSB
STREG + 4	Cylinder number LSB
STREG + 5	Sector number
STREG + 6	User defined byte
STREG + 7	ID field checksum
STREG + 8	Number of cylinders read
STREG + 9 to STREG + E	Not used

TABLE 4-15. DISK/TAPE COMMANDS

Command	Opcode	Definition
BURST DISK BACK-UP	B0	Copy designated cylinders from disk to tape
TRANSPARENT DISK BACK-UP*	B4	
BURST DISK RESTORE	B8	Load designated cylinders from tape to disk
TRANSPARENT DISK RESTORE*	BC	

*Indicates a transparent command.

TABLE 4-16. DISK/TAPE COMMAND COMPLETION CODES

Code	Definition
00 (Hex)	Good execution
01	Good execution; seek retry after seek fault error
02	Good execution; seek retry after cylinder mismatch error
04	Good execution; data retry after ID checksum error
05	Good execution; data retry after CRC error
08	Tape Read operation complete, terminated by file mark
20	Disk drive not present
21	Seek fault
22	Cylinder mismatch error
24	ID checksum error
25	Data CRC error
28	Head Write Protected
31	Invalid disk drive (other than 0 or 1)
33	Invalid cylinder
36	Invalid drive ID
37	Invalid sector size (other than 128, 256, or 512 bytes)
38	Disk (Cylinder Max.) Overrun
60	Tape drive not present
61	Tape cartridge write protected
62	File Mark Detected
63	Block in error not found
65	Hard Data error
68	Cartridge not in place
69	End of Tape
6A	Tape not online
6C	Bottom of Tape
6D	Illegal Command (to tape drive from formatter)
70	Invalid Command with Tape Write Mode set
71	Invalid Command with Tape Read Mode set
73	Tape not available (currently being used in transparent command)

BURST DISK BACK-UP Sequence

1. Assert IMTBUSY.
2. Check for valid disk drive select number.
3. Check for drive on-line and up to speed.
   3.1 Issue drive select and sequence up commands if necessary.
4. Check for valid cylinder number.
   4.1 Issue seek if necessary.
5. Check for valid tape drive select number.
6. Search disk for first sector on start cylinder.
7. Perform disk back-up.
8. Update STREG, remove IMTBUSY and assert CDCPT.

4.6.2 TRANSPARENT DISK BACK-UP Command (Transparent)

INPUTS:

		Command Byte
CMDREG + 0	BIT	7 6 5 4 3 2 1 0
	(B4) Hex VALUE	1 0 1 1 0 1 0 0
CMDREG + 1		Drive select number (0 or 1)
CMDREG + 2		Starting cylinder number MSB
CMDREG + 3		Starting cylinder number LSB
CMDREG + 4 to CMDREG + 6		Not used
CMDREG + 7		Number of read retries if disk error occurs
CMDREG + 8		Number of consecutive cylinders to read
CMDREG + 9		Transparent Command Priority (1 to 9)
CMDREG + A to CMDREG + D		Not used
CMDREG + E		Tape Drive Select (0 to 3 Hex)

OUTPUTS:

STREG + 0	Formatter status byte
STREG + 1	Formatter completion code
STREG + 2	Sync byte from ID field of last sector read
STREG + 3	Head + cylinder number MSB

STREG + 4	Cylinder number LSB
STREG + 5	Sector number
STREG + 6	User defined byte
STREG + 7	ID field checksum
STREG + 8	Number of cylinders read from disk
STREG + 9 to STREG + E	Not used

TRANSPARENT DISK BACK-UP Sequence

1. Remove IMTBUSY and assert XPBUSY.
2. Check for valid disk drive select number.
3. Check for drive on-line and up to speed.
   3.1 Issue drive select and sequence up commands if necessary.
4. Check for valid head and cylinder number.
   4.1 Issue seek if necessary.
5. Check for valid tape drive select number.
6. Search disk for first sector on start cylinder.
7. Perform single cylinder back-up.
   7.1 Check for direct command issued. If so,
   7.2 Save transparent command.
   7.3 Execute direct command.
   7.4 Resume transparent command.
8. Repeat steps 6 - 7 until backup complete.
9. Load status into XP status buffer.
10. Remove EXPBUSY and assert XPCPT.

4.6.3 BURST DISK RESTORE Command (Direct)

INPUTS:

		Command Byte
	BIT	7 6 5 4 3 2 1 0
CMDREG + 0	(B8) Hex VALUE	1 0 1 1 1 0 0 0

CMDREG + 1	Drive select number (0 or 1)
CMDREG + 2	Cylinder number MSB
CMDREG + 3	Cylinder number LSB
CMDREG + 4	Not used
CMDREG + 5	Not used
CMDREG + 6	ID Field User defined byte
CMDREG + 7	Number of retries if disk error
CMDREG + 8	Number of consecutive cylinders to write
CMDREG + 9 to CMDREG + D	Not used
CMDREG + E	Tape Drive Select (0 to 3 Hex)

OUTPUTS:

STREG + 0	Formatter status byte
STREG + 1	Formatter completion code
STREG + 2	Sync byte from ID field of last sector written
STREG + 3	Head + cylinder number MSB
STREG + 4	Cylinder number LSB
STREG + 5	Sector number
STREG + 6	User defined byte
STREG + 7	ID field checksum
STREG + 8	Number of cylinders written
STREG + 9 to STREG + E	Tape status bytes (see Tables 4-10 and 4-11) (only if file mark encountered at end of tape read)

BURST DISK RESTORE Sequence

1. Assert IMTBUSY.
2. Check for drive on-line and up to speed.
   2.1 Issue drive select and sequence up commands, necessary.
3. Check for valid head and cylinder number.
   3.1 Issue seek if necessary.
4. Check for valid tape drive select number.
5. Read first data block from tape.
6. Perform disk restore.
7. If file mark encountered at end of tape operation then read tape status.
8. Update STREG, remove IMTBUSY and assert CDCPT.

4.6.4 TRANSPARENT DISK RESTORE Command (Transparent)

INPUTS:

		Command Byte
	BIT	7 6 5 4 3 2 1 0
CMDREG + 0	(BC) Hex VALUE	1 0 1 1 1 1 0 0

CMDREG + 1	Drive select number (0 or 1)
CMDREG + 2	Starting cylinder number MSB
CMDREG + 3	Starting cylinder number LSB
CMDREG + 4	Not used
CMDREG + 5	Not used
CMDREG + 6	ID field user defined byte
CMDREG + 7	Not used
CMDREG + 8	Number of consecutive cylinders to write
CMDREG + 9	Transparent Command Priority (1 to 9)
CMDREG + A to CMDREG + D	Not used
CMDREG + E	Tape Drive Select (0 to 3 Hex)

OUTPUTS:

STREG + 0	Formatter status byte
STREG + 1	Formatter completion code
STREG + 2	Sync byte from ID field of last sector read
STREG + 3	Head + cylinder number MSB
STREG + 4	Cylinder number LSB
STREG + 5	Sector number
STREG + 6	User defined byte
STREG + 7	ID field checksum
STREG + 8	Number of cylinders written

STREG + 9 to STREG + E	Tape status bytes (see Tables 4-10 and 4-11) (only if file mark encountered at end of tape read)

TRANSPARENT DISK RESTORE Sequence

1. Remove IMTBUSY and assert XPBUSY.
2. Check for valid disk drive select number.
3. Check for drive on-line and up to speed.
   3-1. Issue drive select and sequence up commands if necessary.
4. Check for valid head and cylinder number.
   4-1. Issue seek if necessary.
5. Check for valid tape drive select number.
6. Perform single cylinder restore.
   6.1 Check for direct command issued. If so,
   6.2 Save transparent command.
   6.3 Execute direct command.
   6.4 Resume transparent command.
7. Resume steps 5 - 6 until disk restore is complete.
8. If file mark encountered at end of tape read operation then read tape status.
9. Load status into XP status buffer.
10. Remove XPBUSY and assert XPCPT.

4.7 SYSTEM COMMANDS

The T series formatter has four system commands. The NO-OP command simply confirms controller-host communication. With the COMMAND ACKNOWLEDGE command the host can clear the CDCPT signal. The READ TRANSPARENT COMMAND STATUS (RXS) command serves to update the host on the status of a transparent command by loading the transparent command status into the CSR. The TRANSPARENT COMMAND ACKNOWLEDGE instruction resets the XPCPT flag. Table 4-17 summarizes the system commands.

The system commands have no execution parameters, so all of the CMDREG is unused except the command byte CMDREG+0. Similarly, the system commands themselves have no status parameters. NO-OP and COMMAND ACKNOWLEDGE will return zeros in STREG+2 through STREG+E, while the RXS command will return status information for the current transparent command.

Table 4-18 summarizes the command completion codes for the system commands. COMMAND ACKNOWLEDGE and TRANSPARENT COMMAND ACKNOWLEDGE will return a F0 (Hex) if there is no current command awaiting acknowledgement. Otherwise, COMMAND ACKNOWLEDGE and TRANSPARENT COMMAND ACKNOWLEDGE will return a 00 (Hex). RXS will return the completion code for the current transparent command.

TABLE 4-18. SYSTEM COMMAND COMPLETION CODES

Code (Hex)	Definition
00	Good execution
*18	Rewind; Position Command in Progress
*19	Erase Tape; Position Command in Progress
*1A	Retension; Position Command in Progress
*1B	Advance File Mark Command in Progress
*1C	Transparent Backup in Progress
*1D	Transparent Restore in Progress
30	Invalid controller command
80	Controller reset has occurred
F0	No command awaiting acknowledgement
F4	Transparent command not complete
F8	No Transparent command in progress

*Codes returned by Read Transparent Status only.

4.7.1 NO-OP Command (Direct)

INPUTS:

		Command Byte
CMDREG + 0	BIT	7 6 5 4 3 2 1 0
(00) Hex	VALUE	0 0 0 0 0 0 0 0
CMDREG + 1 to CMDREG + F	Not used	

TABLE 4-17. SYSTEM COMMANDS

Command	Opcode (Hex)	Definition
NO-OP	00	No operation
COMMAND ACKNOWLEDGE	C0	Clear CDCPT signal
READ TRANSPARENT COMMAND STATUS	C1	Load transparent command status bytes into STREG and return status to host
TRANSPARENT COMMAND COMPLETION ACKNOWLEDGE	C2	Acknowledge transparent command completion Clear CDCPT, XPCTP

OUTPUTS:

STREG + 0	Formatter status byte
STREG + 1	Formatter completion code
STREG + 2 to STREG + F	Not used

NO-OP Sequence

1. Assert IMTBUSY.
2. Update STREG.
3. Remove IMTBUSY and assert CDCPT.

4.7.2 COMMAND ACKNOWLEDGE Command (Direct)

INPUTS:

		Command Byte
	BIT	7 6 5 4 3 2 1 0
CMDREG + 0 (C0) Hex	VALUE	1 1 0 0 0 0 0 0

CMDREG + 1 to CMDREG + F	Not used

OUTPUTS:

STREG + 0	Formatter status byte
STREG + 1	Formatter completion code
STREG + 2 to STREG + F	Not used

COMMAND ACKNOWLEDGE sequence

1. Remove CDCPT.

4.7.3 READ TRANSPARENT COMMAND STATUS Command (Direct)

INPUTS:

		Command Byte
	BIT	7 6 5 4 3 2 1 0
CMDREG + 0 (C1) Hex	VALUE	1 1 0 0 0 0 0 1

CMDREG + 1 to CMDREG + F	Not used

OUTPUTS:

STREG + 0	Formatter status byte
STREG + 1	Formatter completion code (See Table 4-18 for transparent command in progress codes)
STREG + 2 to STREG + F	See STREG definition for transparent command in execution

READ TRANSPARENT COMMAND STATUS Sequence

1. Assert IMTBUSY.
2. Load STREG from XP status buffer.
3. Remove IMTBUSY

4.7.4 TRANSPARENT COMMAND ACKNOWLEDGE Command (Direct)

INPUTS:

		Command Byte
	BIT	7 6 5 4 3 2 1 0
CMDREG + 0 (C2) Hex	VALUE	1 1 0 0 0 0 1 0

CMDREG + 1 to CMDREG + F	Not used

OUTPUTS:

STREG + 0	Formatter status byte
STREG + 1	Formatter completion code
STREG + 2 to STREG + F	Not used

TRANSPARENT COMMAND ACKNOWLEDGE Sequence

1. Remove XPCPT.

4.8 TEST COMMANDS

The formatter provides the host with five test commands to verify correct operation of the Command & Status registers and FIFO's. These commands, along with the regular formatter Instruction set allow the host to quickly isolate potential problems. The following sections define each of the four test commands.

4.8.1 MEMORY DUMP Command (Direct)

INPUTS:

		Command Byte
	BIT	7 6 5 4 3 2 1 0
CMDREG + 0 (01) Hex	VALUE	0 0 0 0 0 0 0 1

CMDREG + 1 to CMDREG + F	Not used

OUTPUTS:

STREG + 0	Formatter status byte
STREG + 1	Formatter completion code
STREG + 2 to STREG + F	Not used

MEMORY DUMP Sequence

1. Assert IMTBUSY.
2. Move Z80 RAM into FIFO #0.
3. Set up for 1K byte transfer to host from FIFO #0.
4. Assert DTRQST/.
5. After host has read 1K byte remove DTRQST/.
6. Remove IMTBUSY.

4.8.2 HOST FIFO FILL Command (Direct)

INPUTS:

		Command Byte
CMDREG + 0	BIT	7 6 5 4 3 2 1 0
(02) Hex	VALUE	0 0 0 0 0 0 1 0

CMDREG + 1	FIFO # to fill (0 to 2 hex)
CMDREG + 2	Fill Byte
CMDREG + 3 to CMDREG + F	Not used

OUTPUTS:

STREG + 0	Formatter status byte
STREG + 1	Formatter completion code
STREG + 2 to STREG + F	Not used

HOST FIFO FILL Sequence

1. Assert IMTBUSY.
2. Fill FIFO selected in CMDREG + 1 with the character presented in CMDREG + 2.
3. Set up for a full FIFO transfer of data to host.
4. Assert DTRQST/.
5. After host has read full FIFO remove DTRQST/.
6. Clear IMTBUSY.

4.8.3 HOST FIFO ACCESS TEST Command (Direct)

INPUTS:

		Command Byte
CMDREG + 0	BIT	7 6 5 4 3 2 1 0
(03) Hex	VALUE	0 0 0 0 0 0 1 1

CMDREG + 1 — FIFO operation command

Bit 7	Bits 7 – 4 form a nibble (N) which determines the number of 128 byte blocks to transfer to/from the host. The number of blocks is computed with the equation 10 Hex – N.
Bit 6	
Bit 5	
Bit 4	
Bit 3	Set to 0.
Bit 2	Data transfer direction 0 = Host read data from FIFO 1 = Host write data to FIFO
Bit 1	Target FIFO # (00, 01, 10) Bit 0 = LSB
Bit 0	

CMDREG + 2 to CMDREG + F	Not used

OUTPUTS:

STREG + 0	Formatter status byte
STREG + 1	Formatter completion code
STREG + 2 to STREG + F	Not used

HOST FIFO ACCESS TEST Sequence

1. Assert IMTBUSY.
2. Set up transfer requested by CMDREG + 1.
3. Assert DTRQST/.
4. After host has completed transfer remove DTRQST/.
5. Remove IMTBUSY.

4.8.4 COMMAND/STATUS REGISTER TEST Command (Direct)

INPUTS:

		Command Byte
CMDREG + 0	BIT	7 6 5 4 3 2 1 0
(04) Hex	VALUE	0 0 0 0 0 1 0 0

CMDREG + 1	Status Register Fill Byte
CMDREG + 2 to CMDREG + F	Not used

OUTPUTS:

STREG + 0	Formatter status byte
STREG + 1 to STREG + E	Filled with byte presented in CMDREG + 1

COMMAND/STATUS REGISTER TEST sequence

1. Assert IMTBUSY.
2. Copy contents to CMDREG + 1 into STREG + 1 through STREG + E.
3. Remove IMTBUSY.

4.8.5 FORMATTER IDENTIFICATION Command (Direct)

INPUTS:

		Command Byte
CMDREG + 0	BIT	7 6 5 4 3 2 1 0
(08) Hex	VALUE	0 0 0 0 1 0 0 0

CMDREG + 1 to CMDREG + F	Not used

OUTPUTS:

STREG + 0	Formatter status byte
STREG + 1	Formatter Completion Code (00 Hex)
STREG + 2	Disk Drive Manufacturer 20 = Marksman
STREG + 3	Tape Drive Manufacturer 10 = Archive or Cipher
STREG + 4	Formatter Model (FIFO K byte size) 12 = (2K byte FIFOs)
STREG + 5	Software Version 00 → FF
STREG + 6 to STREG + E	Not used

GYPSY IDENTIFICATION Sequence

1. Assert IMTBUSY.
2. Copy contents of Status Register from PROM.
3. Remove IMTBUSY.

SECTION 5
MARKSMAN DISK INTERFACE

5.0 GENERAL

The T series formatter will support both the standard Marksman interface (J2) used on the M20, M40, M80 and M160 drives, and an extended Marksman interface (J4). The extended interface mates with a Driver/ Receiver board mounted in the secondary disk drive, and is capable of driving a cable length of 10 feet.

One formatter will control up to two drives, in any combination of models. The formatter automatically determines the drive model for each drive as selected and generates the necessary "drive unique" parameters necessary to control it.

5.1 STANDARD MARKSMAN INTERFACE CONNECTOR

Connector J2 on the formatter board is the standard Marksman dual-in-line 40 pin connector. Table 5-1 lists the functions on J2. For specific information regarding the Marksman signals, the user should consult the Marksman Performance Specification (P/N 76220-9XX).

TABLE 5-1. MARKSMAN CONNECTOR PIN ASSIGNMENTS

J2 Pin	Signal	J2 Pin	Signal
1	CACK0	2	RST0/
3	CBUS00	3	CBUS10
5		6	CREQ0
7	GND	8	CBUS20
9	CBUS60	10	CBUS70
11	CBUS50	12	CBUS30
13		14	IDX0
15	GND	16	DRDY0
17	CRDY0	18	GND
19	IMHZ0	20	GND
21	SEC0	22	GND
23	CSTAT0	24	GND
25	WRTCLK0	26	GND
27	NRZIN0	28	GND
29	RDCLK0	30	GND
31	GROUND	32	GND
33	WRTGT0/	34	GND
35	RDGATE0	36	
37	RDGATE0	38	
39	WRTUSF0	40	CBUS40

TABLE 5-2. SECONDARY DRIVE INTERFACE CONNECTOR

J4 Pin	Signal	J4 Pin	Signal
1	CBUS01	2	GND
3	CBUS11	4	GND
5	CBUS21	6	GND
7	CBUS31	8	GND
9	CBUS41	10	GND
11	CBUS51	12	GND
13	CBUS61	14	GND
15	CBUS71	16	GND
17	CACK1B	18	GND
19	DRDY1B	20	GND
21	WRTUSF1B	22	GND
23	CRDY1B	24	GND
25	CSTAT1B	26	GND
27	CREQ1B	28	GND
29	RST1B	30	GND
31	RDGAT1B	32	GND
33	WRTGT1B/	34	GND
35	INDX1B	36	GND
37	SEC1B	38	GND
39	GND	40	WRCK1
41	WRCK1/	42	GND
43	RDCK1	44	RDCK1/
45	GND	46	WRDT1
47	WRDT1/	48	GND
49	RDDT1	50	RDDT1/

5.2 SECTOR SIZE

The sector size switches on the currently selected Marksman drive dictate the formatter sector size. Thus, each drive connected to the Controller can have a different sector size. The user must be careful to keep track of the current sector size, because the formatter will not terminate a disk command until the host has accessed the FIFO for the correct number of bytes. The user can select 128, 256 or 512 byte sectors using the switches provided on the Marksman drive. 1024 byte sector sizes are not currently provided for.

5.3 DISK DEFECT MAP

The T series formatter supports automatic defective media recovery when the DEFECT MAP jumper is shorted. When the defect map is enabled, the formatter will

load the drive defect map (upon selection of the drive) and check for bad sectors prior to each read or write operation. If the host requests a bad sector, the formatter will look up the alternate sector assigned to the defective sector and automatically operate on the alternate sector. The formatter internally holds the defect maps of both disk drives. This allows use of differing drives for direct and transparent commands without extensive loading of different defect maps.

The host generates the alternate sector map during a disk format operation. For more detailed information about defective media recovery, refer to Appendix B. If the defect map is disabled, or the controller cannot locate a defect map on the drive, it will perform all read and write operations to the commanded sector.

SECTION 6
ARCHIVE AND CIPHER TAPE INTERFACE

6.0 GENERAL

The formatter is compatible with the standard ARCHIVE interface used on the ARCHIVE and Cipher streaming cartridge tape controller, which works with the 3020B and 9020B Sidewinder tape drives. The formatter interfaces to one ARCHIVE tape controller, which can control up to four tape drives.

6.1 ARCHIVE CONNECTOR

Connector J3 on the formatter is the standard ARCHIVE dual-in-line 50 pin connector. Table 6-1 lists the functions on the ARCHIVE/CIPHER tape connector J3. For specific information regarding signals, the user should consult the appropriate product specification supplied with the tape drives. The drive circuitry onboard the formatter can drive up to 15 feet of cable, measured from formatter plug to the connector on the last tape drive in the daisy chain.

6.2 DATA BLOCK SIZE

The data block size for ARCHIVE/CIPHER tape drives is 512 bytes. The formatter does the conversion between disk sector sizes and tape blocks for all disk/tape transfers. When the host accesses the tape directly, it must use a 2K byte data block size, because the formatter will request 2K bytes of data. This 2K byte block will be transferred to the buffer in the tape drive in 512 byte blocks.

6.3 TAPE POSITIONING

Proper tape positioning is critical for successful disk back-up and restore operations. Because the user has the capability to remove and insert tape cartridges, the formatter cannot know the position of a newly inserted cartridge. Therefore, a good practice is to command a rewind immediately upon insertion of the cartridge into the tape drive, after which the formatter will keep track of tape position until the tape is removed. From Beginning of Tape (BOT) the user can position the tape at the beginning of any file with the ADVANCE FILE MARK(S) command. But it is the responsibility of the user to know the location of particular files on the tape.

TABLE 6-1. ARCHIVE CONNECTOR PIN ASSIGNMENTS

Pin	Signal	Pin	Signal
1	GROUND	2	RESERVED
3	GROUND	4	RESERVED
5	GROUND	6	RESERVED
7	GROUND	8	RESERVED
9	GROUND	10	RESERVED
11	GROUND	12	DBUS7/
13	GROUND	14	DBUS6/
15	GROUND	16	DBUS5/
17	GROUND	18	DBUS4/
19	GROUND	20	DBUS3/
21	GROUND	22	DBUS2/
23	GROUND	24	DBUS1/
25	GROUND	26	DBUS0/
27	GROUND	28	ONLINE/
29	GROUND	30	REQUEST/
31	GROUND	32	RESET/
33	GROUND	34	XFER/
35	GROUND	36	ACK/
37	GROUND	38	READY/
39	GROUND	40	EXCPTN/
41	GROUND	42	DIRC/
43	GROUND	44	RESERVED
45	GROUND	46	RESERVED
47	GROUND	48	RESERVED
49	GROUND	50	RESERVED

SECTION 7
DEI TAPE INTERFACE

7.0 GENERAL

A second version of the formatter is compatible with the DEI interface used on the DEI streaming cartridge tape controller, which works with the 7130, 7190, 7230, and 7290 model tape drives. The T series formatter interfaces to one DEI tape controller, which can control up to four tape drives.

7.1 DEI CONNECTOR

Connector J3 on the formatter is the standard DEI dual-in-line 50 pin connector. Table 6-2 lists the functions on the DEI tape connector J3. For specific information regarding the DEI signals, the user should consult the product specification supplied with the DEI tape drives. The drive circuitry on-board the formatter can drive up to 15 feet of cable, measured from the formatter plug to the connector on the last tape drive in the daisy chain.

7.2 DATA BLOCK SIZE

The data block size for DEI tape drives is 952 bytes. The formatter does the conversion between disk sector sizes and tape blocks for all disk/tape transfers. When the host accesses the tape directly, it must use a 2K byte data block size, because the formatter will automatically set up for 2K byte data blocks from the host, then group the data to conform to the DEI block size.

7.3 TAPE POSITIONING

Proper tape positioning is critical for successful disk back-up and restore operations. Because the user has the capability to remove and insert tape cartridges, the DEI tape drive will automatically rewind a newly inserted cartridge. From Beginning of Tape (BOT) the user can position the tape at the beginning of any file with the ADVANCE FILE MARK(S) command. But it is the responsibility of the user to know the location of particular files on the tape.

TABLE 7-1. DEI CONNECTOR PIN ASSIGNMENTS

Pin	Signal	Pin	Signal
1	CSB/	2	GROUND
3	US2/	4	GROUND
5	US1/	6	GROUND
7	CB0/	8	GROUND
9	CB1/	10	GROUND
11	SLD/	12	GROUND
13	TMD/	14	GROUND
15	ERF/	16	GROUND
17	EOD/	18	GROUND
19	ROM/	20	GROUND
21	EOM/	22	GROUND
23	URY/	24	GROUND
25	DIE/	26	GROUND
27	DAT0/	28	GROUND
29	DAT1/	30	GROUND
31	DAT2/	32	GROUND
33	DAT3/	34	GROUND
35	DAT4/	36	GROUND
37	DAT5/	38	GROUND
39	DAT6/	40	GROUND
41	DAT7/	42	GROUND
43	RESERVED	44	GROUND
45	DSB/	46	GROUND
47	DRY/	48	GROUND
49	RESERVED	50	GROUND

APPENDIX A
MARKSMAN DISK DRIVE SPECIFICATIONS

A.1 UNFORMATTED DRIVE SPECIFICATIONS

Table A-1 presents the basic media parameters of the MARKSMAN Winchester disk product line. Acceptable head and cylinder parameters for the formatter are taken from this table. The maximum acceptable head number is equal to the number of heads – 1 (the first head is referenced as head no. 0). The maximum acceptable cylinder number is equal to (no. of data cylinders – 1). Again the first cylinder is referenced as no. 0.

A.2 FORMATTED DRIVE PARAMETERS

The data storage capability of a given drive is a function of the sector size chosen by the user. The T series supports sector sizes of 128, 256 or 512 bytes/sector. Table A-2 shows the total formatted data capacity of each MARKSMAN drive using hard sectoring with the embedded ID format of the T series. This table also shows the number of sectors/track as a function of the drive type and sector size. The maximum acceptable sector number is equal to the number of sectors/track – 1 (the first sector is referenced as sector no. 0).

For additional information on a MARKSMAN product the user should refer to the MARKSMAN Performance Specification, P/N 76220-9XX.

TABLE A-1. MARKSMAN MEDIA PARAMETERS

Model Number	No. Of Data Heads	No. Of Cylinders		Total No. Of Data Tracks	Bytes Per Track	Unformatted Drive Capacity
		Data	Alt			
M20	4	209	01	836	24,000	20.16 MB
M40	8	209	01	1672	24,000	40.32 MB
M80	6	559	01	3354	24,000	80.64 MB
M160	6	836	01	5016	32,000	162.20 MB

TABLE A-2. FORMATTED DATA CAPACITY

Model Number	Sector Size					
	128 Bytes		256 Bytes		512 Bytes	
	Sectors Per Track	Drive Cap. (FMT)	Sectors Per Track	Drive Cap. (FMT)	Sectors Per Track	Drive Cap. (FMT)
M20	121	12.94 MB	74	15.83 MB	40	17.12 MB
M40	121	25.89 MB	74	31.67 MB	40	34.24 MB
M80	121	51.94 MB	74	63.53 MB	40	68.68 MB
M160	163	104.65 MB	97	124.55 MB	54	138.60 MB

APPENDIX B
T SERIES DISK FORMAT

B.1 SECTOR FORMAT

The T Series uses a hard sector format with an embedded identification field. Figure B-1 shows an example of this format along with the MARKSMAN index and sector mark signals. Table B-1 explains each of the bytes in the ID field. The sector select switches on the MARKSMAN must be set up to match the allowed data bytes/sector discussed in Table A-2.

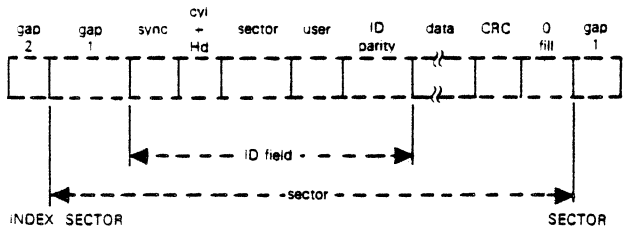


Figure B-1. Hard Sector Embedded I.D. Format

The formatter uses the ID field of the preceding sector to locate the desired sector for a write or read operation. If an ID field is unreadable due to defective media both the defective sector and following sector will be flagged as defective and should be assigned to an alternate track.

B.2 DEFECTIVE MEDIA RECOVERY

The formatter firmware provides the user with two options with respect to defective media recovery. When the DEFECT MAP ENABLE jumper is shorted (enabled) the formatter will use the defect map stored on the jumper to correct for media defeciencies. With the MAP option disabled (jumper open), the controller will simply read and write on the commanded sectors with the results determined by the media.

TABLE B-1 BLOCK DEFINITION

Block	Contents	# of Bytes
Gap 2	Zeros to fill to end of track	Determined by drive type
	Start of physical track Start of 1st physical sector	
Gap 1	Zeros	39 bytes
Sync	ID sync	1 byte
CYL (MSB)	H3 H2 H1 H0 0 C10 C9 C8	1 byte
(LSB)	C7 C6 C5 C4 C3 C2 C1 C0	1 byte
Sector	0 S6 S5 S4 S3 S2 S1 S0	1 byte
User	User specified byte	1 byte
I.D. parity	Parity of 5 previous bytes	1 byte
Data	Data Storage	128, 256 or 512 bytes
CRC	Data CRC	2 bytes
Fill	Zeros	19 bytes
Minimum total bytes/sector		194, 322 or 578
	Start of 2nd physical sector	
Gap 1	Zeros	39 bytes

B.2.1 Defective Media Map

The defective media map used by the formatter is duplicated on the last cylinder on all heads in the first three sectors of the cylinder. The map is written independent of sector size. Only the first 128 bytes of each sector are used by the controller, the remainder of the sector is fill characters.

Table B-2 lists definitions of variables and shows how the formatter uses the map to locate a defective sector and its alternate sector. The host must follow the format specified in Table B-2 when creating a defect map during a drive format operation.

B.2.2. Defective Map Creation Logic

In order for the formatter to correct for defective media it must use the defect map discussed in the preceding paragraphs. The creation of this map is the proper format is the responsibility of the host and in usually done during a drive format routine. While performing a format operation, the formatter informs the host of the defective sectors so that the host can assign alternate sectors to them. Figure B-2 is a flow diagram of a format routine which will format a disk and create the alternate sector map for that disc. The routine writes the defect map at the last cylinder of the drive in sectors 0, 1, and 2 on each head. The redundancy allows for the potential problem of a bad sector at the location the defect map is being written. During normal operation if the formatter cannot read the defect map on head 0 it tries head 1 and continues to the next head if necessary. In the event it cannot read any of the maps, it defaults to the DEFECT MAP DISABLED mode where all read and writes are at the commanded sector.

ROUTINE: DRVFMT (Drive Format)

FUNCTION:
To perform a disk format operation and create the drive defect map.

OPERATION:

DRVFMT first performs a head-by-head format of the drive. For a given head, the "Format Track with Track CRC Verify" command will format each track individually. As each track is formatted DRVFMT keeps track of the defects reported by the controller. The format used to keep track of the defects is identical to the Defect Map format which will eventually be written onto the drive.

After formatting the entire drive, DRVFMT begins to assign alternate sector locations to the defective sectors. As it performs the assignment it makes sure that the defective sectors on the alternate tracks are not assigned as alternate sectors.

Once the defect map is complete with all alternate sectors assigned DRVFMT writes the Defect Map on the last cylinder of the drive at sector locations 0, 1, and 2 on each hand.

DRVFMT: Definition of Variables

DFM(i,j)	Defect map; i = 1 to 64 DFM(i,1) = Head + cylinder MSB of bad sector DFM(i,2) = Cylinder LSB of bad sector DFM(i,3) = Sector no. of bad sector DFM(i,4) = Alternate track number for alternate sector (actual cylinder no. on drive = Max Cylinder No. – alternate track no.) DFM(i,5) = Sector no. of alternate sector
ASCEN	Number of alternate sectors assigned ASCEN range = 0 to 63
HDNS(j)	Head Defect number; Indicates number of defective sectors associated with head no.j
BADCT	Bad sector counter used to keep track of number of bad sectors for current head
BUFFER+i	Formatter FIFO memory i range = 0 to 511
CMDREG+i	Formatter Command register i range = 0 to 7 CMDREG+0 = command byte CMDREG+1 = drive select no. CMDREG+2 = cylinder MSB CMDREG+3 = cylinder LSB CMDREG+4 = head select no. CMDREG+5 = format fill byte (for format command) CMDREG+5 = sector no. (for write command) CMDREG+6 = user defined byte for sector ID
STATUS+i	Controller status register i range = 0 to 7 STATUS+1 = Completion code STATUS+8 = Number of bad sectors detected
ATHEAD	Alternate head count variable
DEFNUM	Defect number counter variable
J	Counter variable used for head no. range = 0 to HDMAX
HDMAX	Maximum head no. for drive being formatted
I	Counter variable used for cylinder no. range = 0 to CYLMAX

TABLE B-2. DEFECTIVE MEDIA MEMORY MAP

Defective Media Map Sector #	Relative Byte #	Variable	Comments
0	0	DFM (1,1)	DEFS(i,n) represent a defective sector. The format for the five bytes of DEFS(i,n) is as follows:
0	1	DFM(1,2)	i = 1 to 64
0	2	DEM(1,3)	n = 1: Head and cylinder MSB information of defect 0 H2 H1 H0 0 C10 C9 C8
0	3	DFM(1,4)	
0	4	DFM(1,5)	n = 2: Cylinder LSB information of defect C7 C6 C5 C4 C3 C2 C1 C0
0	5 → 9	DFM(2,n)	n = 3: Sector # of defect S7 S6 S5 S4 S3 S2 S1 S0
0	10 → 14	DFM(3,n)	
			n = 4: Alternate head on defect replacement cyl. 0 0 0 0 H3 H2 H1 H0
0	. . .	.	
			n = 5: Sector # of alternate sector
0	120 → 124	DFM(25,n)	S7 S6 S5 S4 S3 S2 S1 S0
0	125 → 127	DFM(26,n)	
1	0 → 1	DFM(26,n)	
1	2 → 6	DFM(27,n)	
1	7 → 11	DFM(28,n)	
1	.	.	
1	.	.	
1	122 → 126	DFM(51,n)	
1	127	DFM(52,n)	
2	0 → 3	DFM(52,n)	
2	4 → 8	DFM(53,n)	
2	9 → 13	DFM(54,n)	
.	. .	.	
.	. .	.	
2	59 → 63	DFM(64,n)	
2	64	ASECN	Number of alternate sectors assigned. range: 0 → 128

2	65 → 71
	(reserved space for 7 heads)
		HDNS(j)	Head Defect Number of defective sector. Indicates the number of defective sectors associated with head #j.
			j = 0 → HDMAX
			HDNS(j) = 0; no bad sectors associated with head #j

2	72 → 73 DB E4		Valid defect map character DBE4 (Hex)

CYLMAX	Maximum cylinder no. for drive being formatted (defect sector replacements are contained on this cylinder)	SECMAX	Maximum sector number for drive type
K	Counter variable used to unload defect sector numbers from BUFFER	MAP(i)	MAP(0) = 1st 128 bytes of Defect Map MAP(1) = 2nd 128 bytes of Defect Map MAP(2) = 3rd 128 bytes of Defect Map

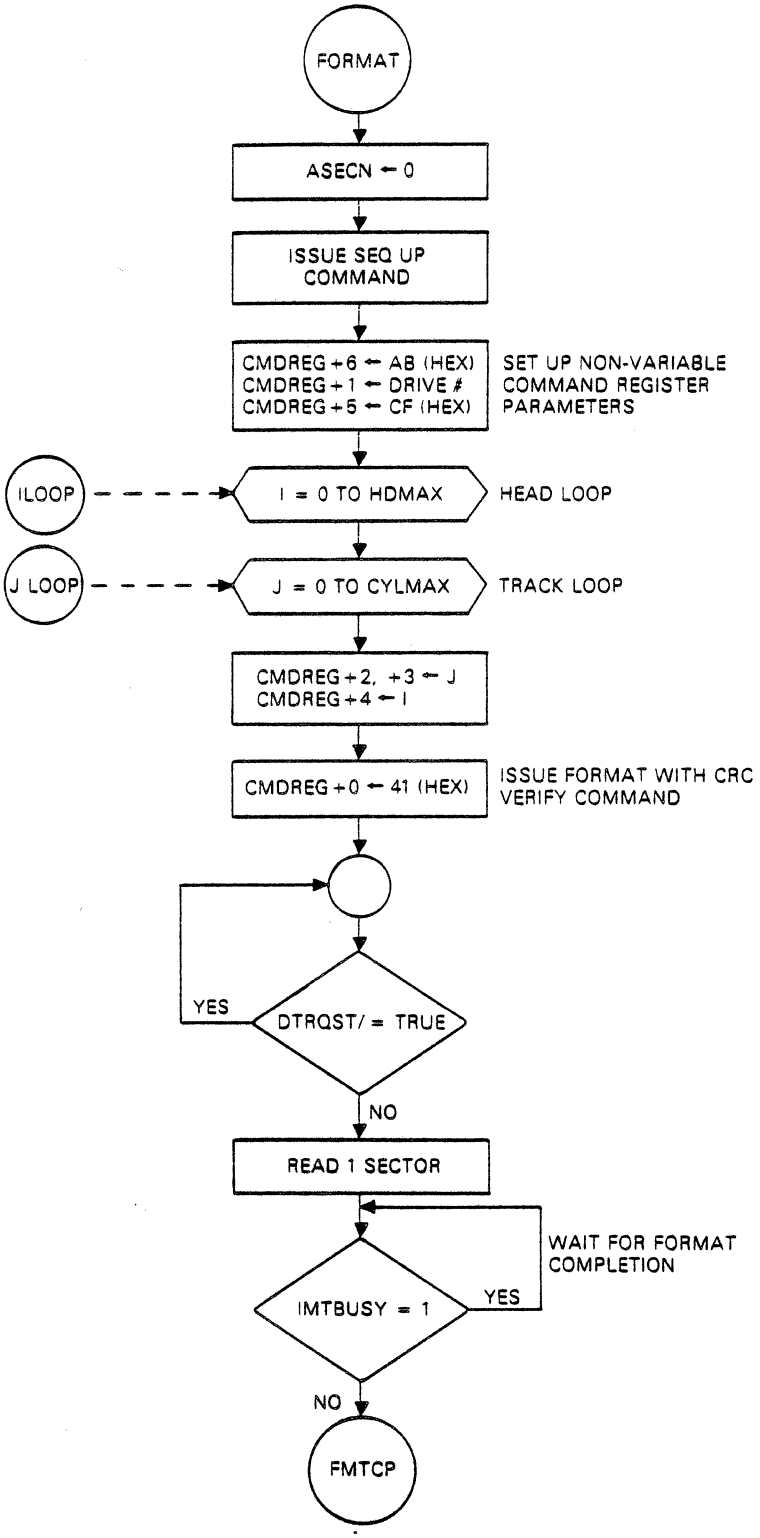


Figure B-2. DRVFMT Flowchart (Page 1 of 6)

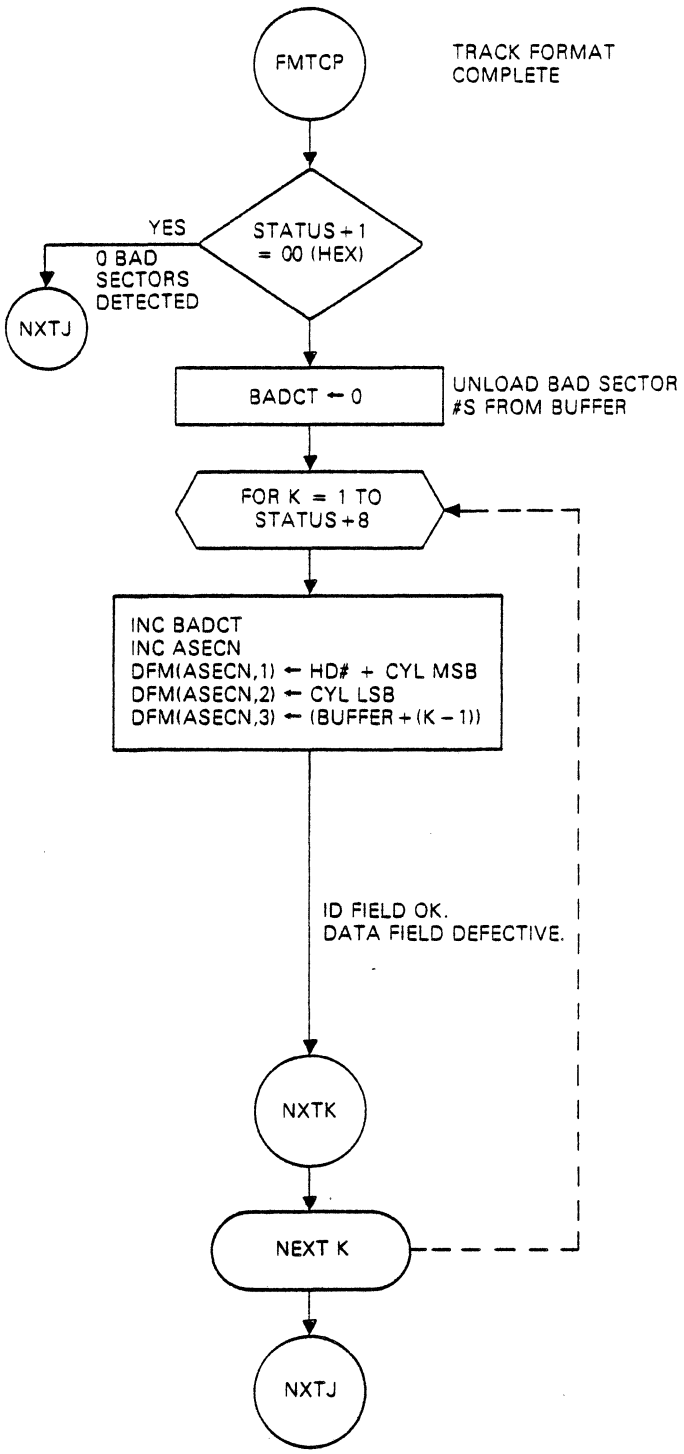


Figure B-2. DRVFMT Flowchart (Page 2 of 6)

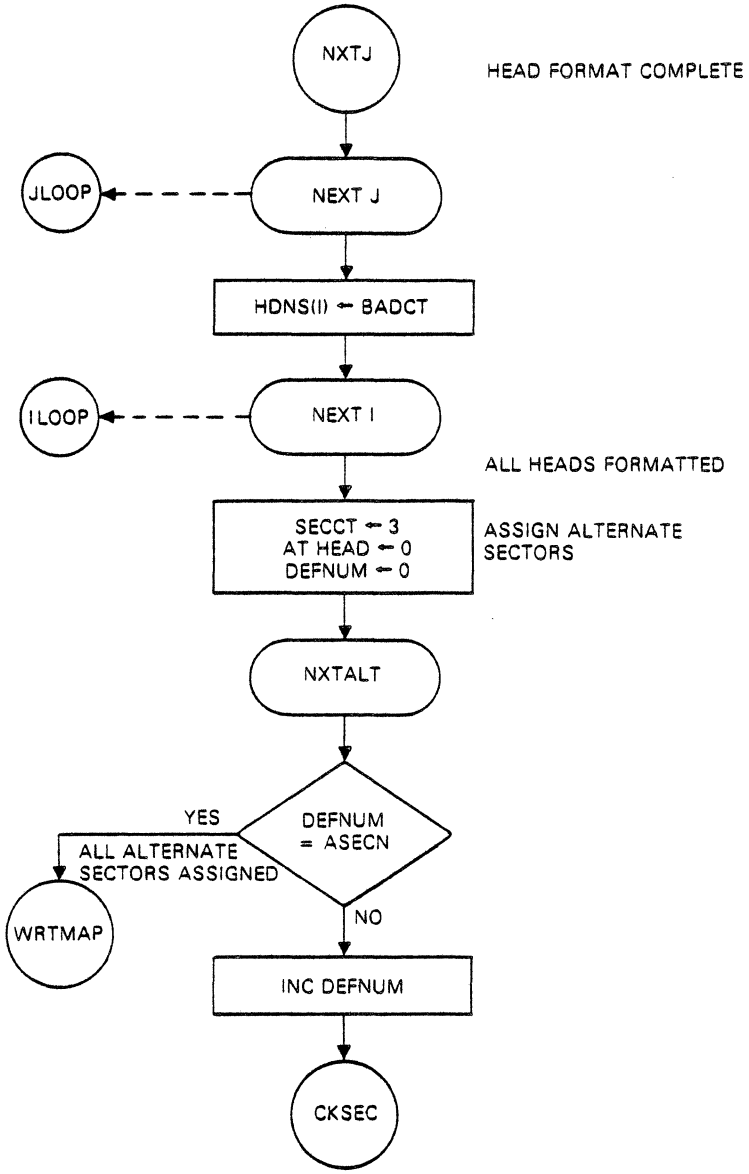


Figure B-2. DRVFMT Flowchart (Page 3 of 6)

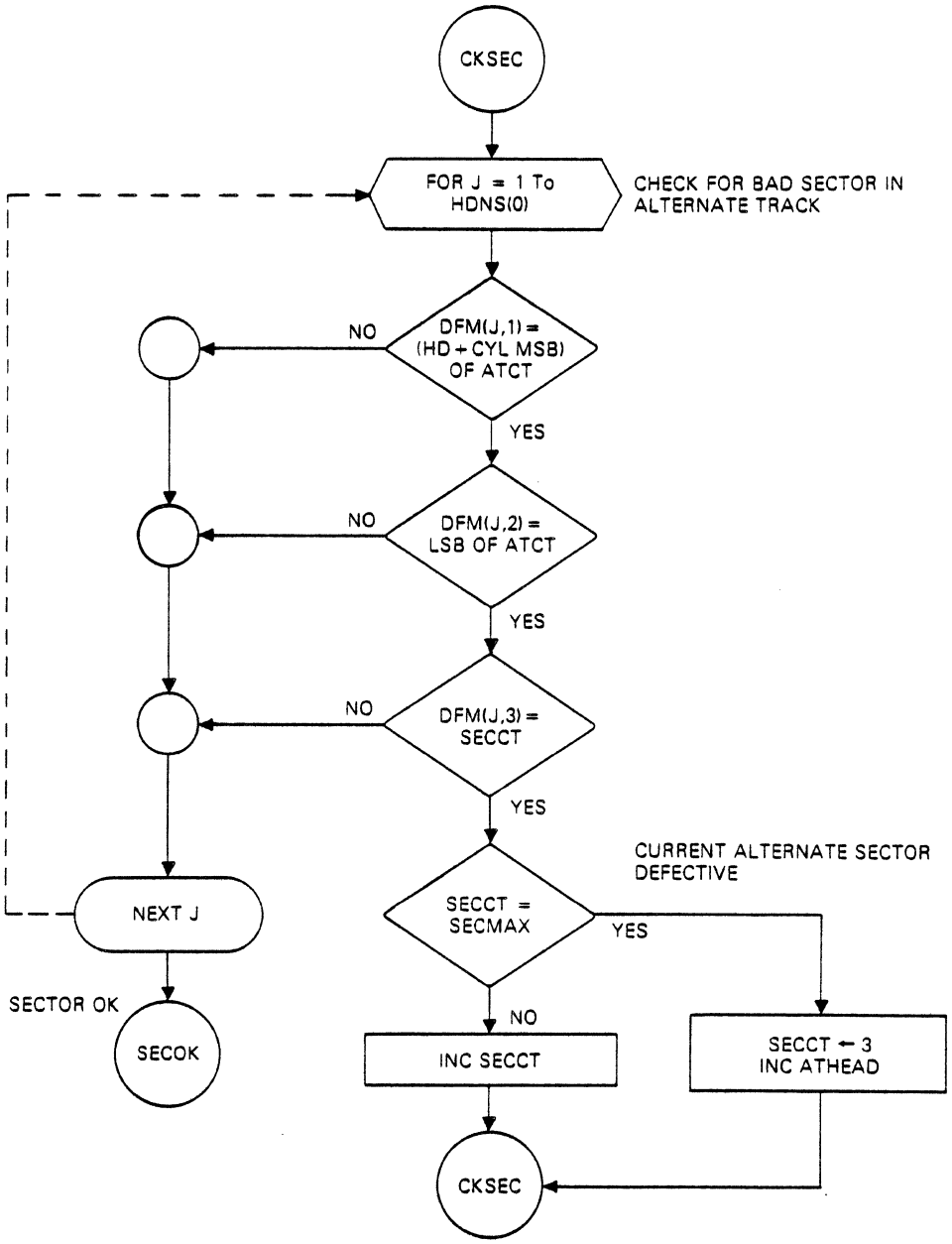


Figure B-2. DRVFMT Flowchart (Page 4 of 6)

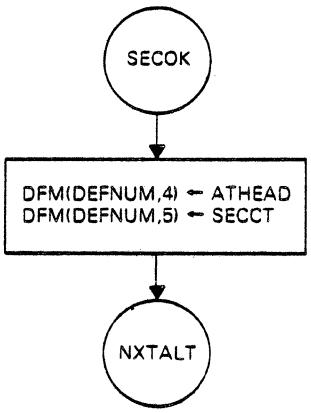


Figure B-2. DRVFMT Flowchart (Page 5 of 6)

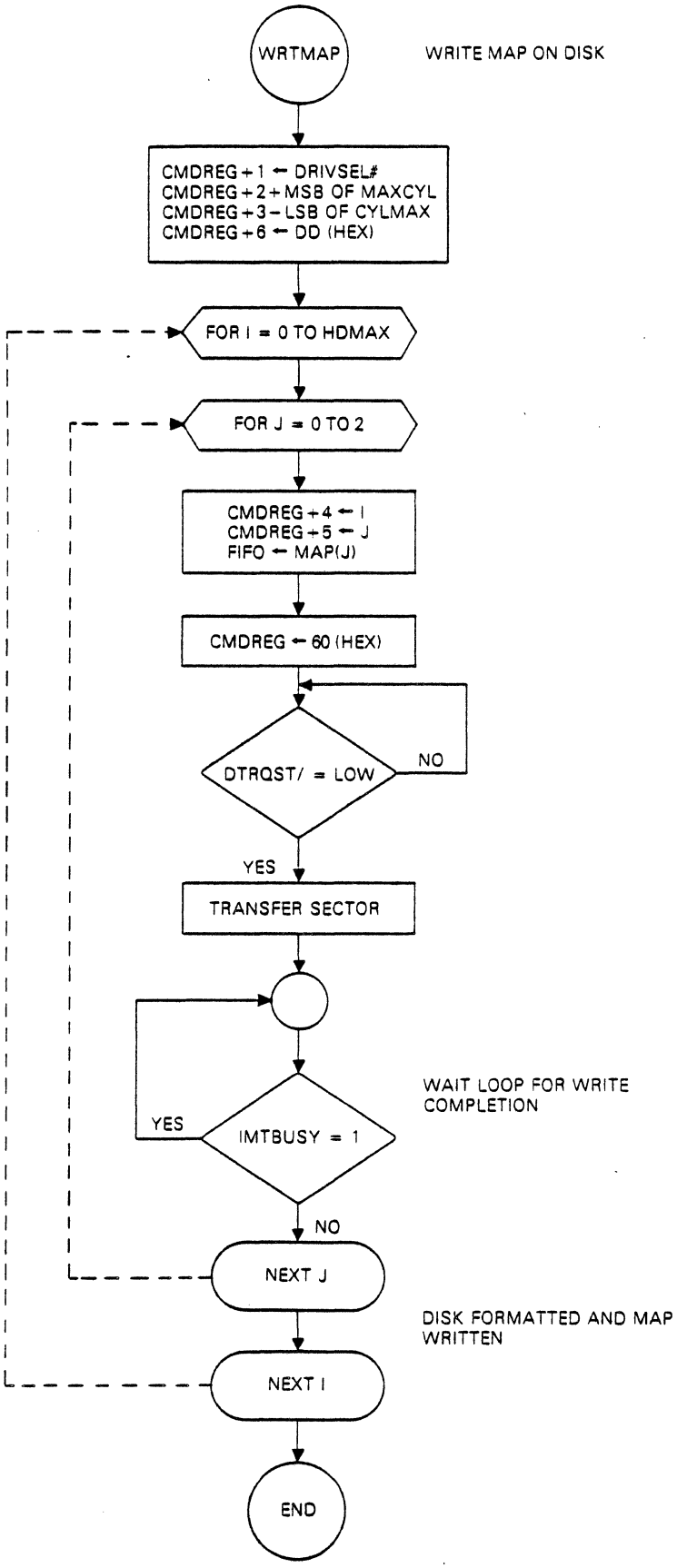


Figure B-2. DRVFMT Flowchart (Page 6 of 6)

READER'S COMMENT FORM

Date: _____

Title of Publication: _____

Part Number of Publication: _____

Your comments help us produce better publications. Please use this post-paid form to submit corrections or suggestions. Include specific page references if appropriate.

☐ Check here and include your name and address if you want copies of this form for future use.

Space is available on the other side of this page for additional comments.

Thank you for your cooperation.

Staple

Fold Fold

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST CLASS PERMIT NO. 19 ANAHEIM, CA

Postage Will Be Paid By Addressee

Century Data Systems
P.O. Box 3056
Anaheim, CA 92803

Attention: Technical Publications

Fold Fold

Additional Comments:

Century Data Systems

A Xerox Company

U.S. Sales Offices

13773 N. Central Expressway, Suite 1314
Dallas, TX 75243
Phone 214/699-1682

5629 FM 1969W, Suite 312
Houston, TX 77069
Phone 713/893-6123

5 Main Street
Cold Spring Harbor, NY 11724
Phone 516/367-4101
TWX 516-367-4292

3375 Scott Blvd., Suite 440
Santa Clara, CA 95051
Phone 408/727-2833

2925 Cardinal Drive, Suite E
P.O. Box 3867
Vero Beach, FL 32960
Phone 305/567-5700
TWX 910-591-1154

6425 So. Pennsylvania
Lansing, MI 48910
Phone 517/393-8088

69 Hickory Drive
Waltham, MA 02154
Phone 617/890-1670

571 N. Poplar, Suite H
Orange, CA 92668
Phone 714/978-0800

2020 Algonquin Road, Suite 307
Schaumburg, IL 60193
Phone 312/397-1007

10 Mechanic Steet, Suite C
Red Bank, NJ 07701
Phone 201/741-1407

International Sales Offices

Haingasse 13-15
6380 Bad Homburg VDH
West Germany

Cory House, 7th Floor
The Ring
Bracknell, Berks U.K. RG12 1ER
England

Century Data Systems, Inc.
1270 North Kraemer, Anaheim, California 92806
P.O. Box 3056, Anaheim, California 92803
For information Phone (714) 632-7500
Telex 655437

CD 1024 5M 1179 Printed in USA
Xerox® is a trademark of XEROX CORPORATION